IN-09
61448
P-66

NASA Contractor Report 189556

(NASA-CR-189556) THE NASA LANGLEY RESEARCH CENTER 0.3-METER TRANSONIC CRYOGENIC TUNNEL MICROCOMPUTER CONTROLLER SOURCE CODE (Vigyan Research Associates) 66 p CSCL 14B

N92-15077

Unclas
G3/09 0061448

The NASA Langley Research Center 0.3-Meter Transonic Cryogenic Tunnel Microcomputer Controller Source Code

W. Allen Kilgore and S. Balakrishna

ViGYAN Inc.
Hampton, Virginia 23666-1325

Contract NAS1-18585
December 1991

National Aeronautics and Space Administration

Langley Research Center
Hampton, Virginia 23665-5225

Contents **Page**

Introduction 1

Controller Changes 2

Set Tunnel Operation Limits 2

Display of Input Limits 3

Improve Screen Layout 3

Aerodynamic Chord Input 4

Tunnel Operating Envelope Safety Catches 4

Identify Causes of an "Emergency Stop" 4

Improve Organization of Source Code 4

Improve Cooldown Efficiency 4

Improve "On" Set Point Flags 5

Rewrite Source Code 6

References 9

Appendices 10

A. Controller Source Code Listing. 10

B. Controller Simulator Source Code Listing. 30

C. Source Code Variables. 49

Limits Input Keyboard Commands. 49

Controller Input Keyboard Commands. 49

Program Variables. 50

List of Figures

1. Maximum - Minimum Limits Display. 7

2. Typical Controller Display. 8

Introduction

The NASA Langley Research Center 0.3-m Transonic Cryogenic Tunnel (TCT) has been operational since 1973. It is a closed circuit fan driven pressure tunnel in which the heat generated by the fan operation is cancelled by injection of liquid nitrogen (LN_2) into the tunnel. The injected LN_2 evaporates into the tunnel and cools the tunnel resident nitrogen gas (GN_2) mass. The gaseous mass buildup and pressure variation caused by LN_2 injection is corrected by controlled discharge of warmer GN_2 tunnel gases. The gas temperature, pressure, and flow Mach number of the tunnel can be controlled by adjusting the LN_2 injection and the GN_2 discharge for a given fan speed.

In 1988 a microcomputer was installed at the 0.3-m TCT to control the temperature, pressure, LN_2 back pressure, fan speed, Mach number and Reynolds number. This system replaced the existing microprocessor control system. This new controller system consists of a PC AT clone with a 12 MHz CPU, an EGA video, a hard disk, a floppy disk, and an 8 channel digital to analog converter. This Controller was designed specifically for ease in modification and operational use. This system has now been operating for several thousand hours providing a safe and efficient control of the tunnel. (Ref. 1 and 2)

Since the Controller has been in use, the staff has gained an extensive knowledge of the Controller and its features. Through the operational use of the Controller several recommendations for improvements have been suggested. Many of these suggestions have been incorporated into the Controller and are described later in this document. The control laws and operations of the Controller have not changed from the original Controller.

The purpose of this document is to provide the source code listing of the 0.3-m TCT Controller which is in Appendix A. Also included is the source code listing for the tunnel simulator (Appendix B) and a complete listing of the variables used in each source code (Appendix C).

Controller Changes

Described below are the changes made to the original Controller. A reason and a description of the change are given for a better understanding of the improvements made to the Controller. Changes to the Controller software should be performed by a person who fully understands the BASIC language. This should be performed only with the knowledge and approval of the Facility Safety Head of the 0.3-m TCT.

Set Tunnel Operation Limits

A requested change was to have the ability for the user to set the tunnel operating limits. An example of this is to have the user set a maximum pressure limit of 50 psia. This limit is below the tunnel maximum operating pressure of 88 psia and is therefore an acceptable limit. With the user set 50 psia limit, the user would not be allowed to input pressures above 50 psia while operating the Controller. This ability to have the user set the tunnel operating limits required an addition to the beginning of the original Controller.

The Controller is now designed so that when the Controller is first executed a Maximum - Minimum Limits program is run which allows the user to set the operating limits of the tunnel. The user set limits must be within the operating envelope of the tunnel. This Maximum - Minimum Limits monitor display is shown in figure 1. This display is self-explanatory and operates using similar keystrokes as the Controller. (Ref. 2) The Maximum - Minimum Limits program allows the user to set a minimum temperature, maximum pressure, maximum fan speed, maximum Mach number, and maximum Reynolds number. The user can either start the Controller with the tunnel limits or the user set limits.

To set the pressure limit the user would strike the "**P**" key on the keyboard. This would cause a "**,psia**" to appear on the screen in the Set Maximum column. The user keys in the desired pressure limit and strikes the "Enter" key. If the desired limit is unusable, the program will warn the user of the problem and will not allow the Controller to be started. The other limits may be set by striking the appropriate **T**, **N**, **M**, or **R** keys on the keyboard. The Maximum - Minimum Limits program has several safety features which prevent the user from setting unacceptable or unsafe limits.

These user set limits are only enforced when the tunnel is in the automatic control

mode of the variable. For example, if the user set limit for pressure is 50 psia and is operating in automatic pressure control, the tunnel Controller will not allow pressure inputs above 50 psia. If the user is operating in automatic Reynolds number control, the user again will not be able to exceed the set Reynolds number limit. If the user is operating in manual GN_2 exhaust valve control, the pressure can exceed the 50 psia limit but not the tunnel pressure limit of 88 psia.

Because the Reynolds number control is obtained through pressure control, the Reynolds number will be restricted by the pressure limit of 50 psia and, therefore limits the Reynolds number even though a user set limit for Reynolds number is not set. This Reynolds number limitation can be easily overcome by lowering the tunnel gas temperature.

Display of Input Limits

With the ability of the user to set the tunnel operating limits the user needs to know these limits. These limits are now displayed in the bottom two rows of the Controller display screen in the section titled **LIMITS** as shown in figure 2. The limit ranges for each variable are displayed within their corresponding control loop columns.

Improve Screen Layout

The Controller display layout shown in figure 2 is nearly identical to the original Controller except for **LIMITS** displayed at the bottom two rows of the screen. Improvements have been made in color and spacing for the visual warnings for an "Emergency Stop". Also, displays are now centered in their perspective control loop columns.

The Reynolds number display has been changed to allow Reynolds numbers of over 99 million. Before, when the Reynolds number reached 100 million, the display showed a % symbol indicating a display overflow.

The RPM/MACH LOOP now clears the old set point when changing from manual to automatic control. The Pt/Re LOOP also clears the old set point when changing from automatic pressure control to automatic Reynolds number control.

Aerodynamic Chord Input

In the original Controller chord length input was in meters. This has been changed so that the chord input is now in inches. This change has been made to improve ease of use for the Controller.

Tunnel Operating Envelope Safety Catches

Included in the Controller are safety catches which will safely stop the tunnel by an "Emergency Stop" if the tunnel limits for pressure or normal fan speed are exceeded. In the original Controller it was possible to over pressurize the tunnel when operating with manual GN_2 exhaust valve control. These safety catches are additional safety features which are duplicated in several other tunnel safety devices separate from the Controller.

Identify Causes of an Emergency Stop

The Controller "Emergency Stop" procedure brings the tunnel to a safe condition. The original Controller displayed only "Emergency Stop" and if the problem was a sensor failure, "Sensor Failure" was also displayed. This was frustrating for the user not knowing what had caused the "Emergency Stop". This "Emergency Stop" condition now displays the cause of the "Emergency Stop" and identifies the failed sensor during a "Sensor Failure". Causes of an "Emergency Stop" are displayed as "Pressure Limit", "Fan RPM Limit", or "Sensor Failure". During a "Sensor Failure" the failed sensors are identified as "Gas Temperature", "Wall Temperature", "Total Pressure", "Static Pressure", and "Screen Pressure". These displays will allow the 0.3-m TCT staff to correct tunnel problems quickly.

Improved Organization of Source Code

The source code has now been divided into logical sections which pertain to certain functions of the Controller. These divisions are designed to allow for an easier understanding of the source code. Some sections have been rearranged to reduce the number of lines and improve the Controller execution times.

Improved Cooldown Efficiency

The automatic cooldown of the Controller is one of its best features. It allows for a safe, automatic, and efficient cooldown of the tunnel to any desired *temperature set point.* The temperature control loop achieves this cooldown by using a *"use" temperature set point* which is based on an acceptable temperature difference between the tunnel wall and gas.

During tunnel operations the *temperature set point* is changed frequently by the user and the Controller always uses the previous cycle *"use" temperature set point* as a starting point to reach the desired *temperature set point.* This works well until the tunnel is already cool and a user chooses a warmer *temperature set point* and then changes back to a cold *temperature set point.* Because the Controller uses the previous cycle *"use" temperature set point* to start cooling, it can take several cycles before the *"use" temperature set point* reaches the actual tunnel gas temperature. This provides a period during a cooldown in which the tunnel is actually warming up waiting for the *"use" temperature set point* to reach the actual tunnel gas temperature.

This problem has been corrected by setting the *"use" temperature set point* previous cycle value, **STP**, to the tunnel gas temperature, **TT**. This occurs only during a new temperature input to cooldown the tunnel.

Improved "On" Set Point Flags

The "On" set point flags are black bands which are shown when the tunnel is within a given tolerance of a desired set point. These "On" set point flags work well to indicate quickly and easily when the tunnel had reached a desired condition. However, there was confusion about the temperature and Reynolds number "On" set point flags.

In the original Controller the temperature "On" set point flag was displayed when the absolute difference between the gas temperature, **TT**, and the *"use" temperature set point*, **ST**, was less then or equal to 0.3. This condition for displaying the "On" set point flag allowed the flag to be displayed when the tunnel was not near the desired *temperature set point*, **ST1**. This caused some confusion in determining if the tunnel was actually "On" temperature set point. To remove this confusion the "On" set point flag for temperature is now displayed when the absolute difference between the tunnel temperature, **TT**, and the desired *temperature set point*, **ST1** is less than or equal to 0.3.

When operating in the Reynolds number control loop the user chooses a desired Reynolds number and a pressure is calculated to give this desired Reynolds number. The confusion occurs because Reynolds number control is still through the pressure control loop. The "On" set point flag in the original Controller is for pressure only. This can cause the "On" set point flag to be "On" when in the Reynolds number control loop while there is still a large difference between the desired Reynolds number set point, **SRE**, and the tunnel Reynolds number, **RE**. The solution to eliminate this confusion was to make a separate condition for the "On" set point flag for use when in the Reynolds number control loop. This has been accomplished by the addition of the logic flag **FL7**. The Reynolds number "On" set point flag is now displayed when the difference between the desired Reynolds number set point, **SRE**, and the tunnel Reynolds number, **RE**, is less than or equal to 0.05.

The controller now has "On" set point flags which display when the conditions of the tunnel are those of the desired user set points.

Rewrite Source Code

The original source code was written in the BASIC language which requires a line number for every line of code used in the program. This line numbering causes some difficulty when trying to add to the program. The source code has been rewritten and compiled using MicroSoft QuickBASIC (Ref. 3). The advantages of using QuickBASIC are that it does not require line numbers for each line of code and all statements are identical to the original source code. Line numbers in QuickBASIC are only required on lines which are called by other lines. This lack of line numbers makes additions or changes much simpler. QuickBASIC also has a BASIC compiler that provides a faster compiled Controller program. The source code listings in Appendix A and B are in QuickBASIC.

0.3-m TUNNEL T-P/R-M CONTROLLER

	LN PUMP AUTO	TEMP LOOP AUTO	Pt/Re LOOP AUTOP	RPM/MACH LOOP AUTO
SET POINT	132.3,psia	200.0,K(Final) 200.0,K(Use)	45.00,Psia ,Miln	0.765 ,Mach ,RPM
PROCESS	132.3,psia	200.0,K-GN2 200.5,K-WALL	45.00,Psia 12.79,Miln	0.765,Mach 4165.,RPM
COMMAND	69.6,%opn	20.1,%opn	36.0,%opn V1 0.0,%opn V2	55.5 ,% Rhst
INPUTS Delete		Temp= ALQ%=	Pres= Ryno= AGv%= Chrd=	Mach= Nrpm=
STATUS		GRAD= 0.0,K/mt SAT= 93.8,K	CHORD= 7.09,in P st= 30.54,psia Del P= 0.000,psi	
LIMITS		77 < T < 340	14.7 < P < 88.0 1.0 < R < 50.0	0 < N < 5600 0.15 < M < 0.995

Figure 2. Typical Controller Display.

Maximum - Minimum Limits for 0.3-m TCT Controller

	Tunnel Maximum	Set Maximum		Set Minimum	Tunnel Minimum
Temperature (kelvin)	340.0	340.0	< T <		Saturation Temperature
Pressure (psia)	88.00		< P <	14.70	Ambient Pressure
Normal Fan Speed (rpm)	5600.		< N <	0.	0.
Mach Number	0.995		< M <	0.150	0.150
Reynolds Number (Miln)	50.00		< R <	1.00	1.00

Accept Tunnel Max./Min. Limits and Start 0.3-m TCT Controller.
Accept Set Max./Min. Limits and Start 0.3-m TCT Controller.

Figure 1. Maximum - Minimum Limits Display.

References

1. Balakrishna, S.; and Kilgore, W. Allen: Microcomputer Based Controller for the Langley 0.3-Meter Transonic Cryogenic Tunnel. NASA CR 181808, March 1989.

2. Balakrishna, S.; and Kilgore, W. Allen: The NASA Langley Research Center 0.3-Meter Transonic Cryogenic Tunnel T-P/Re-M Controller Manual. NASA CR 181868, July 1989.

3. MicroSoft QuickBASIC, Programming in BASIC, Version 4.5. 1988.

4. User Manual for DT 2801 Series, Single Board Analog and Digital I/O System for the IBM Personal Computer. Data Translation, Inc., Eight Edition. October 1988.

Appendix A
Controller Source Code Listing

```
'0.3-m Transonic Cryogenic Tunnel Controller.
        CLS
        CLEAR

'Set up of the Digital to Analog Converter.
        DEFINT I, Z
        DIM E(8), DAC(8)
        ZASE.ADDRESS = &H2EC
        ZOMMAND.REGISTER = ZASE.ADDRESS + 1
        ZTATUS.REGISTER = ZASE.ADDRESS + 1
        ZATA.REGISTER = ZASE.ADDRESS
        ZOMMAND.WAIT = &H4
        WRITE.WAIT = &H2
        READ.WAIT = &H5
        ZERROR = &H2
        ZCLEAR = &H1
        ZADIN = &HC
        ZSTOP = &HF


'Digital to Analog Conversion Board Test.
        OUT &H225, 0
        FOR JJ = 1 TO 300
        BSD = SQR(7)
        NEXT JJ
        AEMP = INP(&H225)
                IF AEMP = 4 THEN 2
        PRINT "Digital to Analog Conversion Problem!": PRINT
        GOTO 10000
2       OUT &H224, &H6

'Program Constants.
        DEL = .1
        CGV = 8
        CH = .18
        CHIN = .18 * .0245
        CLQV = 4
        IW = 39
        K = 1
        KDL = 0!
        KDP = 0!
        KDT = 0!
        KIL = .02
        KIM = 0!
        KIN = .4
        KIP = .05
        KIT = .1
        KMM = .3
        KP = .3
        KPL = .2
        KPM = 4.5
        KPN = .6
        KPP = 1!
        KPT = 1!
```

```
        KT = .04
        MXT = 40
        SLQSC = 17!
        STP = 300!
        XDLP = 1!
        XFRPM = 1280
        XPLQ = 5.103429
        XPP = 1.366
        XPS = 1.366

'Maximum - Minimum Limits Screen Layout.
        COLOR 14, 9
        CLS
        LOCATE 3, 16: PRINT " Maximum - Minimum Limits for 0.3-m TCT Controller "

        COLOR 12, 9
        LOCATE 5, 25: PRINT "Tunnel"
        LOCATE 6, 24: PRINT "Maximun"
        LOCATE 5, 38: PRINT "Set"
        LOCATE 6, 36: PRINT "Maximun"
        LOCATE 5, 60: PRINT "Set"
        LOCATE 6, 58: PRINT "Minimum"
        LOCATE 5, 72: PRINT "Tunnel"
        LOCATE 6, 71: PRINT "Minimum"

        COLOR 7, 9
        LOCATE 7, 3: PRINT "Temperature"
        LOCATE 8, 4: PRINT "(kelvin)"
        LOCATE 10, 3: PRINT "Pressure"
        LOCATE 11, 4: PRINT "(psia)"
        LOCATE 13, 3: PRINT "Normal Fan Speed"
        LOCATE 14, 4: PRINT "(rpm)"
        LOCATE 16, 3: PRINT "Mach Number"
        LOCATE 19, 3: PRINT "Reynolds Number"
        LOCATE 20, 4: PRINT "(Miln)"
        LOCATE 22, 3: PRINT "Accept Tunnel Max./Min. Limits and Start 0.3-m TCT Controller." LOCATE
23, 3: PRINT "Accept Set Max./Min. Limits and Start 0.3-m TCT Controller."

        COLOR 15, 9
        LOCATE 7, 51: PRINT "T"
        LOCATE 7, 49: PRINT CHR$(243)
        LOCATE 7, 53: PRINT CHR$(243)
        LOCATE 10, 51: PRINT "P"
        LOCATE 10, 49: PRINT CHR$(243)
        LOCATE 10, 53: PRINT CHR$(243)
        LOCATE 13, 51: PRINT "N"
        LOCATE 13, 49: PRINT CHR$(243)
        LOCATE 13, 53: PRINT CHR$(243)
        LOCATE 16, 51: PRINT "M"
        LOCATE 16, 49: PRINT CHR$(243)
        LOCATE 16, 53: PRINT CHR$(243)
        LOCATE 19, 51: PRINT "R"
        LOCATE 19, 49: PRINT CHR$(243)
        LOCATE 19, 53: PRINT CHR$(243)

        COLOR 14, 9
        LOCATE 7, 3: PRINT "T"
```

```
        LOCATE 10, 3: PRINT "P"
        LOCATE 13, 3: PRINT "N"
        LOCATE 16, 3: PRINT "M"
        LOCATE 19, 3: PRINT "R"
        LOCATE 22, 3: PRINT "A"
        LOCATE 23, 10: PRINT "S"

'Tunnel Maximum - Minimum Limits.
        MAXT1 = 340!
        MINT1 = 77!
        MAXP1 = 88!
        MINP1 = 14.696
        MAXN1 = 5600!
        MINN1 = 0!
        MAXM1 = .995
        MINM1 = .15
        MAXRE1 = 50!
        MINRE1 = 1!
        MAXCHIN = 15.75
        MINCHIN = .394
        MAXCH = .4
        MINCH = .01
        MAXLQSC = 150!
        MAXSPR = MAXP1 / 14.696
        MINSPR = MINP1 / 14.696

        MAXT = MAXT1
        MINT = MINT1
        MAXP = MAXP1
        MINP = MINP1
        MAXN = MAXN1
        MINN = MINN1
        MAXM = MAXM1
        MINM = MINM1
        MAXRE = MAXRE1
        MINRE = MINRE1

        COLOR 10, 9
        LOCATE 7, 25: PRINT USING "###.#"; MAXT1
        LOCATE 7, 37: PRINT USING "###.#"; MAXT1
        LOCATE 10, 25: PRINT USING "##.##"; MAXP1
        LOCATE 10, 59: PRINT USING "##.##"; MINP1
        LOCATE 13, 25: PRINT USING "####."; MAXN1
        LOCATE 13, 60: PRINT USING "####."; MINN1
        LOCATE 16, 25: PRINT USING "#.###"; MAXM1
        LOCATE 16, 60: PRINT USING "#.###"; MINM1
        LOCATE 19, 25: PRINT USING "##.##"; MAXRE1
        LOCATE 19, 60: PRINT USING "##.##"; MINRE1
        LOCATE 7, 69: PRINT "Saturation"
        LOCATE 8, 69: PRINT "Temperature"
        LOCATE 10, 71: PRINT "Ambient"
        LOCATE 11, 71: PRINT "Pressure"
        LOCATE 13, 73: PRINT USING "####."; MINN1
        LOCATE 16, 73: PRINT USING "#.###"; MINM1
        LOCATE 19, 73: PRINT USING "##.##"; MINRE1
```

```
'Maximum - Minimum Limits Inputs.
5       A$ = INKEY$
                IF A$ = "T" OR A$ = "t" THEN GOSUB 10
                IF A$ = "P" OR A$ = "p" THEN GOSUB 100
                IF A$ = "N" OR A$ = "n" THEN GOSUB 200
                IF A$ = "M" OR A$ = "m" THEN GOSUB 300
                IF A$ = "R" OR A$ = "r" THEN GOSUB 400
                IF A$ = "A" OR A$ = "a" THEN 500
                IF A$ = "S" OR A$ = "s" THEN 600
        GOTO 5

'Temperature Minimum Limit Subroutine.
10      COLOR 15, 9
        BEEP
        LR = 7
        LC = 58
        LOCATE LR, LC + 1: PRINT "      ,K"
        XX = 0
20      B$ = INKEY$
                IF B$ = "D" OR B$ = "d" THEN 30 ELSE 40
30      LOCATE LR, LC + 1: PRINT "          "
        RETURN
40              IF B$ = "0" OR B$ = "1" OR B$ = "2" OR B$ = "3" OR B$ = "4" OR B$ = "5" OR B$ =
                "6" OR B$ = "7" OR B$ = "8" OR B$ = "9" OR B$ = "." THEN 50 ELSE 20
50      XX = XX + 1
                IF XX = 1 THEN H1 = VAL(B$): LOCATE LR, LC + 1: PRINT USING "#"; H1
                IF XX = 2 THEN H2 = VAL(B$): LOCATE LR, LC + 2: PRINT USING "#"; H2
                IF XX = 3 THEN H3 = VAL(B$): LOCATE LR, LC + 3: PRINT USING "#"; H3
                IF XX = 4 THEN LOCATE LR, LC + 4: PRINT CHR$(46)
                IF XX = 5 THEN H5 = VAL(B$): LOCATE LR, LC + 5: PRINT USING "#"; H5
                IF XX = 5 THEN 60 ELSE 20
60      AA = H1 * 100 + H2 * 10 + H3 + H5 / 10
        MINT = AA
                IF MINT >= MAXT1 THEN MINT = MAXT1
                IF MINT <= MINT1 THEN MINT = MINT1
70      C$ = INKEY$
                IF C$ = CHR$(13) THEN 80 ELSE 70
80      BEEP
        COLOR 14, 9
        LOCATE LR, LC + 1: PRINT USING "###.#  "; MINT
        RETURN

'Pressure Maximum Limit Subroutine.
100     COLOR 15, 9
        BEEP
        LC = 36
        LR = 10
        LOCATE LR, LC + 1: PRINT "      ,psia"
        XX = 0
110     B$ = INKEY$
                IF B$ = "D" OR B$ = "d" THEN 120 ELSE 130
120     LOCATE LR, LC + 1: PRINT "    "
        RETURN

130             IF B$ = "0" OR B$ = "1" OR B$ = "2" OR B$ = "3" OR B$ = "4" OR B$ = "5" OR B$ =
                "6" OR B$ = "7" OR B$ = "8" OR B$ = "9" OR B$ = "." THEN 140 ELSE 110
140     XX = XX + 1
```

```
            IF XX = 1 THEN H1 = VAL(B$): LOCATE LR, LC + 1: PRINT USING "#"; H1
            IF XX = 2 THEN H2 = VAL(B$): LOCATE LR, LC + 2: PRINT USING "#"; H2
            IF XX = 4 THEN H3 = VAL(B$): LOCATE LR, LC + 4: PRINT USING "#"; H3
            IF XX = 3 THEN LOCATE LR, LC + 3: PRINT CHR$(46)
            IF XX = 5 THEN H5 = VAL(B$): LOCATE LR, LC + 5: PRINT USING "#"; H5
            IF XX = 5 THEN 150 ELSE 110
150     AA = H1 * 10 + H2 + H3 / 10 + H5 / 100
        MAXP = AA
            IF MAXP <= MINP1 THEN 160 ELSE 170
160     MAXP = MINP1
        COLOR 20, 8: LOCATE LR + 1, 36: PRINT " Warning!  Set Max. = Tunnel Min. "
        FLO = 1
        GOTO 180
170     COLOR 14, 9: LOCATE LR + 1, 36: PRINT "                                    "
        FLO = 0
180         IF MAXP > MAXP1 THEN MAXP = MAXP1
190     C$ = INKEY$
            IF C$ = CHR$(13) THEN 195 ELSE 190
195     BEEP
        COLOR 14, 9
        LOCATE LR, LC + 1: PRINT USING "##.##      "; MAXP
        RETURN

'Normal Fan Speed Maximum Limit Subroutine.
200     COLOR 15, 9
        BEEP
        LC = 36
        LR = 13
        LOCATE LR, LC + 1: PRINT "      ,rpm"
        XX = 0
210     B$ = INKEY$
            IF B$ = "D" OR B$ = "d" THEN 220 ELSE 230
220     LOCATE LR, LC + 1: PRINT "   "
        RETURN
230         IF B$ = "0" OR B$ = "1" OR B$ = "2" OR B$ = "3" OR B$ = "4" OR B$ = "5" OR B$ =
            "6" OR B$ = "7" OR B$ = "8" OR B$ = "9" OR B$ = "." THEN 240 ELSE 210
240     XX = XX + 1
            IF XX = 1 THEN H1 = VAL(B$): LOCATE LR, LC + 1: PRINT USING "#"; H1
            IF XX = 2 THEN H2 = VAL(B$): LOCATE LR, LC + 2: PRINT USING "#"; H2
            IF XX = 3 THEN H3 = VAL(B$): LOCATE LR, LC + 3: PRINT USING "#"; H3
            IF XX = 5 THEN LOCATE LR, LC + 5: PRINT CHR$(46)
            IF XX = 4 THEN H5 = VAL(B$): LOCATE LR, LC + 4: PRINT USING "#"; H5
            IF XX = 5 THEN 250 ELSE 210
250     AA = H1 * 1000 + H2 * 100 + H3 * 10 + H5
        MAXN = AA
            IF MAXN <= MINN1 THEN 260 ELSE 270
260     MAXN = MINN1
        COLOR 20, 8: LOCATE LR + 1, 36: PRINT " Warning!  Set Max. = Tunnel Min. "
        FLO = 1
        GOTO 280
270     COLOR 14, 9: LOCATE LR + 1, 36: PRINT "                                    "
        FLO = 0
280         IF MAXN > MAXN1 THEN MAXN = MAXN1
290     C$ = INKEY$
            IF C$ = CHR$(13) THEN 295 ELSE 290
295     BEEP
        COLOR 14, 9
```

```
        LOCATE LR, LC + 1: PRINT USING "####.      "; MAXN
        RETURN

'Mach Number Maximum Limit Subroutine.
300     COLOR 15, 9
        BEEP
        LC = 36
        LR = 16
        LOCATE LR, LC + 1: PRINT "      ,Mach"
        XX = 0
310     B$ = INKEY$
                IF B$ = "D" OR B$ = "d" THEN 320 ELSE 330
320     LOCATE LR, LC + 1: PRINT "    "
        RETURN
330             IF B$ = "0" OR B$ = "1" OR B$ = "2" OR B$ = "3" OR B$ = "4" OR B$ = "5" OR B$ =
                "6" OR B$ = "7" OR B$ = "8" OR B$ = "9" OR B$ = "." THEN 340 ELSE 310
340     XX = XX + 1
                IF XX = 1 THEN H1 = VAL(B$): LOCATE LR, LC + 1: PRINT USING "#"; H1
                IF XX = 3 THEN H2 = VAL(B$): LOCATE LR, LC + 3: PRINT USING "#"; H2
                IF XX = 4 THEN H3 = VAL(B$): LOCATE LR, LC + 4: PRINT USING "#"; H3
                IF XX = 2 THEN LOCATE LR, LC + 2: PRINT CHR$(46)
                IF XX = 5 THEN H5 = VAL(B$): LOCATE LR, LC + 5: PRINT USING "#"; H5
                IF XX = 5 THEN 350 ELSE 310
350     AA = H1 + H2 / 10 + H3 / 100 + H5 / 1000
        MAXM = AA
                IF MAXM <= MINM1 THEN 360 ELSE 370
360     MAXM = MINM1
        COLOR 20, 8: LOCATE LR + 1, 36: PRINT " Warning!  Set Max. = Tunnel Min. "
        FL0 = 1
        GOTO 380
370     COLOR 14, 9: LOCATE LR + 1, 36: PRINT "                                    "
        FL0 = 0
380             IF MAXM > MAXM1 THEN MAXM = MAXM1
390     C$ = INKEY$
                IF C$ = CHR$(13) THEN 395 ELSE 390
395     BEEP
        COLOR 14, 9
        LOCATE LR, LC + 1: PRINT USING "#.###      "; MAXM
        RETURN

'Reynolds Number Maximum Limit Subroutine.
400     COLOR 15, 9
        BEEP
        LC = 36
        LR = 19
        LOCATE LR, LC + 1: PRINT "      ,Miln"
        XX = 0
410     B$ = INKEY$
                IF B$ = "D" OR B$ = "d" THEN 420 ELSE 430
420     LOCATE LR, LC + 1: PRINT "    "
        RETURN

430             IF B$ = "0" OR B$ = "1" OR B$ = "2" OR B$ = "3" OR B$ = "4" OR B$ = "5" OR B$ =
                "6" OR B$ = "7" OR B$ = "8" OR B$ = "9" OR B$ = "." THEN 440 ELSE 410
440     XX = XX + 1
                IF XX = 1 THEN H1 = VAL(B$): LOCATE LR, LC + 1: PRINT USING "#"; H1
                IF XX = 2 THEN H2 = VAL(B$): LOCATE LR, LC + 2: PRINT USING "#"; H2
```

```
            IF XX = 4 THEN H3 = VAL(B$): LOCATE LR, LC + 4: PRINT USING "#"; H3
            IF XX = 3 THEN LOCATE LR, LC + 3: PRINT CHR$(46)
            IF XX = 5 THEN H5 = VAL(B$): LOCATE LR, LC + 5: PRINT USING "#"; H5
            IF XX = 5 THEN 450 ELSE 410
450     AA = H1 * 10 + H2 + H3 / 10 + H5 / 100
        MAXRE = AA
            IF MAXRE <= MINRE1 THEN 460 ELSE 470
460     MAXRE = MINRE1
        COLOR 20, 8: LOCATE LR + 1, 36: PRINT " Warning!  Set Max. = Tunnel Min. "
        FL0 = 1
        GOTO 480
470     COLOR 14, 9: LOCATE LR + 1, 36: PRINT "                                  "
        FL0 = 0
480         IF MAXRE > MAXRE1 THEN MAXRE = MAXRE1
490     C$ = INKEY$
            IF C$ = CHR$(13) THEN 495 ELSE 490
495     BEEP
        COLOR 14, 9
        LOCATE LR, LC + 1: PRINT USING "##.##      "; MAXRE
        RETURN

'Starts the 0.3-m TCT Controller with Tunnel Maximum - Minimum Limits.
500     XX = 0
        GOTO 800

'Starts the 0.3-m TCT Controller with Set Maximum - Minimum Limits.
600         IF FL0 = 1 THEN GOTO 5
        XX = 0
        MAXSPR = MAXP / 14.696
        MINSPR = MINP / 14.696
        GOTO 800

'0.3-m TCT T-P/Re-M Controller Program.

'Controller Screen Layout.
800     COLOR 0, 0
        CLS
        COLOR 15, 4
        LOCATE 1, 26: PRINT " 0.3-m TUNNEL T-P/R-M CONTROLLER "

        FOR I = 2 TO 25
        COLOR 15, 1
        LOCATE I, 2: PRINT STRING$(10, 0);
            IF I = 4 THEN LOCATE 4, 2: PRINT STRING$(10, 205);
            IF I = 14 THEN LOCATE 14, 2: PRINT STRING$(10, 205);
            IF I = 19 THEN LOCATE 19, 2: PRINT STRING$(10, 205);
            IF I = 23 THEN LOCATE 23, 2: PRINT STRING$(18, 205);

        COLOR 15, 2
        LOCATE I, 12: PRINT STRING$(12, 0);
            IF I = 4 THEN LOCATE 4, 12: PRINT STRING$(12, 205);
            IF I = 14 THEN LOCATE 14, 12: PRINT STRING$(12, 205);
            IF I = 19 THEN LOCATE 19, 12: PRINT STRING$(12, 205);
            IF I = 23 THEN LOCATE 23, 12: PRINT STRING$(18, 205);

        COLOR 15, 3
        LOCATE I, 24: PRINT STRING$(17, 0);
```

```
        IF I = 4 THEN LOCATE 4, 24: PRINT STRING$(17, 205);
        IF I = 14 THEN LOCATE 14, 24: PRINT STRING$(17, 205);
        IF I = 19 THEN LOCATE 19, 24: PRINT STRING$(17, 205);
        IF I = 23 THEN LOCATE 23, 24: PRINT STRING$(18, 205);

COLOR 15, 5
LOCATE I, 41: PRINT STRING$(18, 0);
        IF I = 4 THEN LOCATE 4, 41: PRINT STRING$(18, 205);
        IF I = 14 THEN LOCATE 14, 41: PRINT STRING$(18, 205);
        IF I = 19 THEN LOCATE 19, 41: PRINT STRING$(18, 205);
        IF I = 23 THEN LOCATE 23, 41: PRINT STRING$(18, 205);

COLOR 15, 6
LOCATE I, 59: PRINT STRING$(18, 0);
        IF I = 4 THEN LOCATE 4, 59: PRINT STRING$(18, 205);
        IF I = 14 THEN LOCATE 14, 59: PRINT STRING$(18, 205);
        IF I = 19 THEN LOCATE 19, 59: PRINT STRING$(18, 205):
        IF I = 23 THEN LOCATE 23, 59: PRINT STRING$(18, 205);
NEXT I

COLOR 15, 2
LOCATE 2, 14: PRINT " LN PUMP ": PRINT ;
LOCATE 3, 14: PRINT "  AUTO ": PRINT ;
LOCATE 5, 18: PRINT ",psia": PRINT ;
LOCATE 9, 18: PRINT ",psia": PRINT ;
LOCATE 12, 18: PRINT ",%opn": PRINT ;
LOCATE 15, 12: PRINT " =": PRINT ;
LOCATE 5, 13: PRINT USING "###.#"; SLQSC

COLOR 14, 2
LOCATE 15, 12: PRINT "B": PRINT ;

COLOR 15, 3
LOCATE 2, 28: PRINT "TEMP LOOP": PRINT ;
LOCATE 3, 29: PRINT " MANUAL ": PRINT ;
LOCATE 5, 31: PRINT ",K(Final)": PRINT ;
LOCATE 6, 31: PRINT ",K(Usc)": PRINT ;
LOCATE 9, 31: PRINT ",K-GN2": PRINT ;
LOCATE 10, 31: PRINT ",K-WALL": PRINT ;
LOCATE 12, 31: PRINT ",%opn": PRINT ;
LOCATE 15, 25: PRINT " emp=": PRINT ;
LOCATE 16, 25: PRINT "A Q%=": PRINT ;
LOCATE 20, 25: PRINT "GRAD=       ,K/mt": PRINT ;
LOCATE 21, 25: PRINT " SAT=       ,K": PRINT ;
LOCATE 24, 32: PRINT "T";
LOCATE 24, 30: PRINT CHR$(243);
LOCATE 24, 34: PRINT CHR$(243);
LOCATE 24, 26: PRINT USING "###"; MINT;
LOCATE 24, 36: PRINT USING "###"; MAXT;

COLOR 14, 3
LOCATE 15, 25: PRINT "T": PRINT ;
LOCATE 16, 26: PRINT "L": PRINT ;

COLOR 15, 5
LOCATE 2, 45: PRINT "Pt/Rc LOOP": PRINT ;
LOCATE 3, 46: PRINT " MANUAL ": PRINT ;
```

```
LOCATE 5, 48: PRINT ",Psia": PRINT ;
LOCATE 6, 48: PRINT ",Miln": PRINT ;
LOCATE 20, 42: PRINT "CHORD= 7.09,in": PRINT ;
LOCATE 9, 48: PRINT ",Psia": PRINT ;
LOCATE 21, 42: PRINT "P st=        ,psia": PRINT ;
LOCATE 22, 42: PRINT "Del P=       ,psi": PRINT ;
LOCATE 10, 48: PRINT ",Miln": PRINT ;
LOCATE 12, 48: PRINT ",%opn V1": PRINT ;
LOCATE 13, 48: PRINT ",%opn V2": PRINT ;
LOCATE 15, 43: PRINT " res=": PRINT ;
LOCATE 16, 43: PRINT " yno=": PRINT ;
LOCATE 17, 43: PRINT "A v%=": PRINT ;
LOCATE 18, 43: PRINT " hrd=": PRINT ;
LOCATE 24, 50: PRINT "P";
LOCATE 24, 48: PRINT CHR$(243);
LOCATE 24, 52: PRINT CHR$(243);
LOCATE 24, 43: PRINT USING "##.#"; MINP;
LOCATE 24, 54: PRINT USING "##.#"; MAXP;
LOCATE 25, 50: PRINT "R";
LOCATE 25, 48: PRINT CHR$(243);
LOCATE 25, 52: PRINT CHR$(243);
LOCATE 25, 43: PRINT USING "##.#"; MINRE;
LOCATE 25, 54: PRINT USING "##.#"; MAXRE;

COLOR 14, 5
LOCATE 15, 43: PRINT "P": PRINT ;
LOCATE 16, 43: PRINT "R": PRINT ;
LOCATE 17, 44: PRINT "G": PRINT ;
LOCATE 18, 43: PRINT "C": PRINT ;

COLOR 15, 6
LOCATE 2, 62: PRINT "RPM/MACH LOOP": PRINT
LOCATE 3, 64: PRINT " MANUAL ": PRINT ;
LOCATE 5, 67: PRINT ",Mach": PRINT ;
LOCATE 6, 67: PRINT ",RPM": PRINT ;
LOCATE 9, 67: PRINT ",Mach": PRINT ;
LOCATE 10, 67: PRINT ",RPM": PRINT ;
LOCATE 12, 67: PRINT ",% RhsT": PRINT ;
LOCATE 15, 61: PRINT " ach=": PRINT ;
LOCATE 16, 61: PRINT " rpm=": PRINT ;
LOCATE 24, 68: PRINT "N";
LOCATE 24, 66: PRINT CHR$(243);
LOCATE 24, 70: PRINT CHR$(243);
LOCATE 24, 61: PRINT USING "####"; MINN;
LOCATE 24, 72: PRINT USING "####"; MAXN;
LOCATE 25, 68: PRINT "M";
LOCATE 25, 66: PRINT CHR$(243);
LOCATE 25, 70: PRINT CHR$(243);
LOCATE 25, 61: PRINT USING ".###"; MINM;
LOCATE 25, 72: PRINT USING ".###"; MAXM;

COLOR 14, 6
LOCATE 15, 61: PRINT "M": PRINT ;
LOCATE 16, 61: PRINT "N": PRINT ;

COLOR 15, 1
LOCATE 5, 3: PRINT "SET POINT": PRINT ;
```

```
        LOCATE 9, 3: PRINT "PROCESS": PRINT ;
        LOCATE 12, 3: PRINT "COMMAND": PRINT ;
        LOCATE 15, 3: PRINT "INPUTS": PRINT ;
        LOCATE 20, 3: PRINT "STATUS": PRINT ;
        LOCATE 16, 3: PRINT " elete": PRINT ;
        LOCATE 24, 3: PRINT "LIMITS";

        COLOR 14, 1
        LOCATE 16, 3: PRINT "D": PRINT ;

'Controller Program.
1000    GOSUB 8000

'Conversion of Digital Inputs to Engineering Units.
        PP = E(1) * XPP
        PPUSCS = PP * 14.696
        PS = E(2) * XPS
                IF PS > (PP * .999999) THEN PS = PP * .999999
        PSUSCS = PS * 14.696
                IF E(3) > 1.191 THEN 2080 ELSE 2060
2060    TT = 74.1826 + 105.3 * E(3) - 40.66 * E(3) ^ 2 + 20.54 * E(3) ^ 3 - 5.21 * E(3) ^ 4
        GOTO 2090
2080    TT = 80.678 + 84.52 * E(3) - 12.717 * E(3) ^ 2 + 1.805 * E(3) ^ 3 - .1102 * E(3) ^ 4
2090            IF E(4) > 1.191 THEN 2120 ELSE 2100
2100    TMWL = 74.183 + 105.3 * E(4) - 40.66 * E(4) ^ 2 + 20.54 * E(4) ^ 3 - 5.21 * E(4) ^ 4
        GOTO 2130
2120    TMWL = 80.678 + 84.52 * E(4) - 12.717 * E(4) ^ 2 + 1.805 * E(4) ^ 3 - .1102 * E(4) ^ 4
2130    FRPM = E(5) * XFRPM
        PLQ = E(6) * XPLQ
        PLQUSCS = PLQ * 14.696
        DLP = E(7) * XDLP

        GOSUB 5000

        M = SQR(5 * (PP / PS) ^ .28571 - 5)
        MF = (1 + .2 * M * M)
        KRE = 63714 * CH * M / TT ^ 1.4 / (MF) ^ 2.1
        RE = KRE * PP
        SAT = 50 + 27.34 * PS ^ .296
        SAT1 = SAT * MF
        LDPQ = (PLQ - PP)
                IF LDPQ < .5 THEN LDPQ = .5
        LF = CLQV * .8676 * SQR(LDPQ)
        DRPM = FRPM
                IF DRPM < 100 THEN DRPM = 100
        KTGS = DRPM * SQR(PP) * KT / 3! / TT / LF
        TTMP = TT
                IF TTMP < 80 THEN TTMP = 80
        FKW = 100 * PP * (FRPM / 1000) ^ 2.26 / SQR(TTMP)
        FB = FKW / (121 + TT) / LF
        SPR = SP / 14.696
        SPR1 = SRE / KRE
                IF SPR1 > MAXSPR THEN SPR1 = MAXSPR
                IF SPR1 < MINSPR THEN SPR1 = MINSPR
                IF AUTORE = 1 THEN SPR = SPR1
        SPRU = SPR * 14.696
        COLOR 14, 5
```

```
            IF AUTORE = 1 THEN LOCATE 5, 43: PRINT USING "##.##"; SPRU

'Temperature Control Loop.
            IF IE = 1 THEN 2950
            IF (TMWL - TT) > 1.5 * MXT THEN 2950
            IF FRPM < 580 THEN 2950
      ST = ST1
            IF ST > TT THEN 2530 ELSE 2525
2525        IF ST < (STP - .04) THEN ST = (STP - .04)
2530        IF AUTOT = 1 THEN 2540 ELSE 2850
2540        IF AUTOP = 0 THEN 2610
            IF ABS(PP - SPR) < .15 THEN 2560 ELSE 2570
2560        IF ABS(TT - ST) < 2! THEN 2610 ELSE 2570
2570        IF (PP / TT) > .95 * (SPR / ST) THEN 2610 ELSE 2580
2580        IF TT > ST THEN 2610 ELSE 2590
2590  ST = TT - .02
2610        IF (TMWL - ST - MXT) > 0 THEN 2620 ELSE 2650
2620  ST = TMWL - MXT
      FB = 0
2650        IF ST < SAT1 THEN ST = SAT1
      ET = TT - ST
            IF ABS(ET) < .3 THEN FL1 = 1 ELSE FL1 = 0
            IF ABS(TMWL - TT) < 24 THEN FL2 = 1 ELSE FL2 = 0
            IF ET < -5 THEN FF = 0
            IF ET > 0 THEN FF = 1
      FBF = FB * FF * .8
      RIT = RITM1 + KIT * KTGS * DEL * ET
            IF RIT < -FBF THEN RIT = -FBF
            IF RIT > (1 - FBF) THEN RIT = (1 - FBF)
            IF (TMWL - TT) > MXT / 2 THEN 2750 ELSE 2780
2750  LMT = 1 - (TMWL - TT - MXT / 2) * 2 / MXT
            IF LMT < 1! / LF THEN LMT = 1! / LF
            IF RIT > LMT THEN RIT = LMT
2780  ALQ = KTGS * (KPT * ET + KDT * (ET - 2 * ETM1 + ETM2) / 2 / DEL) + RIT + FBF
      IF ALQ < 0 THEN ALQ = 0
            IF ALQ > 1 THEN ALQ = 1
            IF (TMWL - TT) > MXT / 2 THEN 2820 ELSE 2830
2820        IF ALQ > LMT THEN ALQ = LMT
2830  LCMDS = ALQ * 100
      GOTO 2970
2850        IF LCMDS > 100 THEN LCMDS = 100
      ALQ = LCMDS / 100
            IF (TMWL - TT) > MXT / 2 THEN 2880 ELSE 2910
2880  LMT = 1 - (TMWL - TT - MXT / 2) * 2 / MXT
            IF LMT < 1! / LF THEN LMT = 1! / LF
            IF ALQ > LMT THEN ALQ = LMT
2910  RIT = ALQ
      GOTO 2970
2950  ALQ = 0
      LCMDS = 0
2970        IF ALQ > ALQP + .01 THEN ALQ = ALQP + .01
      DAC(1) = ALQ
      DAC(2) = ALQ

'Pressure - Reynolds Number Control Loop.
      KPGS = 750 * KP / PP / SQR(TT)
      GF1 = 2.725 * CGV * PP / SQR(TT)
```

```
            IF PPUSCS > MAXP1 THEN PIE = 1: GOSUB 6000
            IF IE = 1 THEN 3220
            IF ABS(RE - SRE) < .05 THEN FL7 = 1 ELSE FL7 = 0
            IF AUTOP = 1 THEN 3040 ELSE 3180
3040    EP = PP - SPR
            IF ABS(EP) < .005 THEN FL3 = 1 ELSE FL3 = 0
        RIP = RIPM1 + EP * KPGS * KIP * DEL
            IF RIP < 0 THEN RIP = 0
            IF RIP > 1! THEN RIP = 1!
        AGV1 = KPGS * (KPP * EP + KDP * (EP - 2 * EPM1 + EPM2) / 2 / DEL) + RIP
            IF AGV1 < 0 THEN AGV1 = 0
            IF AGV1 > 1! THEN AGV1 = 1!
            IF AGV1 > .9 THEN AGV2 = AGV2 + .01
            IF AGV1 < .7 THEN AGV2 = AGV2 - .01
            IF AGV2 < 0 THEN AGV2 = 0
            IF AGV2 > 1! THEN AGV2 = 1!
        GCMDS = AGV1 * 100
        GOTO 3240
3180        IF GCMDS > 100 THEN GCMDS = 100
        RIP = GCMDS / 100
        AGV1 = GCMDS / 100
            IF FRPM < 300 THEN LCMDS = 0
        GOTO 3240
3220    AGV1 = 1!
        GCMDS = 100
3240        IF AGV1 < AGV1P - .05 THEN AGV1 = AGV1P - .05
            IF AGV1 > AGV1P + .05 THEN AGV1 = AGV1P + .05
        DAC(4) = AGV1 * .8 + .2
        DAC(5) = AGV2 * .8 + .2

'Fan Speed - Mach Number Control Loop.
            IF IE = 1 THEN 3405 ELSE 3410
3405    NCMDS1 = 0
        GOTO 3550
3410    NCMDS1 = NCMDS
            IF FRPM > MAXN1 THEN NIE = 1: GOSUB 6000
            IF AUTOM = 1 THEN 3422 ELSE 3550
3422        IF FRPM < 580 THEN AUTOM = 0: GOTO 3435
        FKM = 1 - .5 * M
            IF FKM < .71 THEN FKM = .71
        KMC = 520 * SQR(TT) * FKM / PP ^ .035
        KMGS = KMC * KMM
            IF ALQ > .99 THEN 3432 ELSE 3440
3432    AUTOM = 0
        NCMDS = FRPM - 500
            IF NCMDS < 0 THEN NCMDS = 0: BEEP
3435    COLOR 14, 6
        LOCATE 3, 64: PRINT " MANUAL ": PRINT
        LOCATE 6, 62: PRINT USING "####."; NCMDS
        GOTO 3550
3440    EM = SM - M
            IF ABS(EM) < .002 THEN FLA = 1 ELSE FLA = 0
        RIM = RIMM1 + KIM * EM * DEL * KMGS
            IF RIM < -100 THEN RIM = -100
            IF RIM > 100 THEN RIM = 100
        NCMDS1 = KPM * EM * KMGS + RIM + FRPM
            IF NCMDS1 > 5450 THEN NCMDS1 = 5450
```

```
            IF NCMDS1 > MAXN THEN NCMDS1 = MAXN
3550    EN = NCMDS1 - FRPM
          IF ABS(EN) < 10  THEN FLN=1 ELSE FLN=0
            IF ABS(EN) > 100 THEN 3560 ELSE 3570
3560    KEN = 5
        GOTO 3575
3570    KEN = SQR(TT) / 20
3575    RIN = RINM1 + EN * DEL * KIN
            IF RIN > RINM1 + KEN THEN RIN = RINM1 + KEN
            IF RIN < RINM1 - KEN THEN RIN = RINM1 - KEN
            IF RIN < 0 THEN RIN = 0
            IF RIN > 6000 THEN RIN = 6000
        SRPM = EN * KPN + RIN
            IF SRPM > SRPM1 + 5 THEN SRPM = SRPM1 + 5
            IF SRPM < SRPM1 - 5 THEN SRPM = SRPM1 - 5
        SNRPM = SRPM / 7500
            IF SNRPM > 1! THEN SNRPM = 1!
            IF SNRPM < 0! THEN SNRPM = 0!
        DAC(6) = SNRPM

'Fan Speed Band Warning.
            IF FRPM < 3651 THEN 3790 ELSE 3830
3790        IF FRPM > 3549 THEN 3800 ELSE 3830
3800    COLOR 20, 8
        LOCATE 13, 61: PRINT " SPEED BAND ": PRINT
        BEEP
        GOTO 3900
3830    COLOR 14, 6
        LOCATE 13, 61: PRINT "                ": PRINT

'Temperature Gradient Calculation.
3900    IW = IW + 1
            IF IW = 40 THEN 3920 ELSE 3980
3920        IF WLG > 8 THEN WLG = 0
            IF WLG < -8 THEN WLG = 0
        COLOR 14, 3
        LOCATE 20, 30: PRINT USING "###.#"; 12 * WLG
        WLG = 0
        IW = 0
3980    WLG = WLG + TMWL - TMWL1


'Liquid Back Pressure Control Loop.
        SLQ = SLQSC / 14.696
            IF SLQ > 10.2 THEN SLQ = 10.2
        SLQSC = SLQ * 14.696
        ELP = PLQ - SLQ
            IF ABS(ELP) < .4 THEN FL5 = 1 ELSE FL5 = 0
            IF ELP > .15 THEN ELP = .15
            IF ELP < -.15 THEN ELP = -.15
        RIL = RILM1 + ELP * KIL * DEL
            IF RIL < 0. THEN RIL = 0.
            IF RIL > 1! THEN RIL = 1!
        ALN = KPL * ELP + KDL * (ELP - ELPM1) / DEL + RIL
            IF ALN < 0. THEN ALN = 0.
            IF ALN > 1! THEN ALN = 1!
        DAC(3) = ALN
```

```
        DAC(7) = 1!

        GOSUB 9000

'Screen Pressure Drop Warning.
        DLPC = .47 * PP * M * M / (MF) ^ 6 + .02
                IF DLP > DLPC THEN FL6 = 1 ELSE FL6 = 0
                IF M < .25 THEN FL6 = 0
                IF FL6 = 1 THEN BEEP

'Controller Screen Update.
        JD = JD + 1
                IF JD = 3 THEN 4160 ELSE 4700
4160    COLOR 14, 3
        LOCATE 21, 30: PRINT USING "###.#"; SAT1
                IF (FL1 * FL2 * AUTOT) = 1 THEN 4190 ELSE 4200
4190    COLOR 14, 0
4200    LOCATE 8, 26: PRINT STRING$(6, 0)
        COLOR 14, 5
        LOCATE 21, 47: PRINT USING "###.##"; E12 * 20.07474
                IF FL6 = 1 THEN COLOR 30, 5
        LOCATE 22, 48: PRINT USING "##.###"; DLP
                IF AUTORE = 0 THEN 4240
                IF (FL7 * AUTORE) = 1 THEN 4250 ELSE 4260
4240            IF (FL3 * AUTOP) = 1 THEN 4250 ELSE 4260
4250    COLOR 14, 0
4260    LOCATE 8, 43: PRINT STRING$(6, 0)
        COLOR 14, 2
        LOCATE 5, 13: PRINT USING "###.#"; SLQSC
        LOCATE 9, 13: PRINT USING "###.#"; PLQUSCS
        LOCATE 12, 13: PRINT USING "###.#"; (1 - ALN) * 100
                IF FL5 = 1 THEN 4320 ELSE 4330
4320    COLOR 14, 0
4330    LOCATE 8, 13: PRINT STRING$(6, 0)
        COLOR 14, 6
                IF (FL4 * AUTOM) = 1 THEN 4360 ELSE 4370
4360    COLOR 14, 0
4370    LOCATE 8, 62: PRINT STRING$(6, 0)
       IF(FLN-AUTOM)>0 THEN 4374 ELSE 4376
4374  COLOR 14,0
4376  LOCATE 11,62: PRINT STRING$(6,000)
      COLOR 14, 3
                IF AUTOT = 1 THEN LOCATE 6, 26: PRINT USING "###.#"; ST
          LOCATE 9, 26: PRINT USING "###.#"; TT
          LOCATE 10, 26: PRINT USING "###.#"; TMWL
          LQ = ALQ * 100
          LOCATE 12, 26: PRINT USING "###.#"; LQ
          COLOR 14, 5
          LOCATE 9, 42: PRINT USING "###.##"; E11 * 20.07474
          GV1 = AGV1 * 100
          LOCATE 12, 43: PRINT USING "###.#"; GV1
          GV2 = AGV2 * 100
          LOCATE 13, 43: PRINT USING "###.#"; GV2
          LOCATE 10, 42: PRINT USING "###.##"; RE
          COLOR 14, 6
          LOCATE 9, 62: PRINT USING "#.###"; M
          LOCATE 10, 62: PRINT USING "####."; FRPM
```

```
        LOCATE 12, 62: PRINT USING "##.#"; SNRPM * 100
        JD = 0

'Setting Previous Cycle Values.
4700    AGV1P = AGV1
        ALQP = ALQ
        ELPM1 = ELP
        EPM2 = EPM1
        EPM1 = EP
        ETM2 = ETM1
        ETM1 = ET
        RILM1 = RIL
        RIMM1 = RIM
        RINM1 = RIN
        RIPM1 = RIP
        RITM1 = RIT
        SRPM1 = SRPM
        STP = ST
        TMWL1 = TMWL

4900    GOTO 1000

'Controller Input Subroutine.
5000    A$ = INKEY$
                IF A$ = "" THEN 5999
                IF A$ = "D" OR A$ = "d" THEN 5950
                IF XX > 5 THEN XX = 5
                IF A$ = CHR$(13) THEN 5008 ELSE 5009
5008            IF XX = 5 THEN 5600 ELSE 5999
5009            IF J > 0 THEN 5010 ELSE 5014
5010            IF A$ = "0" OR A$ = "1" OR A$ = "2" OR A$ = "3" OR A$ = "4" OR A$ = "5" OR A$
                = "6" OR A$ = "7" OR A$ = "8" OR A$ = "9" OR A$ = "." THEN 5100 ELSE 5999
5014     COLOR 16, 3
                IF A$ = "T" OR A$ = "t" THEN J = 1: LOCATE 15, 36: PRINT ",K": PRINT ;
                IF A$ = "L" OR A$ = "l" THEN J = 5: LOCATE 16, 36: PRINT ",opn": PRINT ;
        COLOR 16, 5
                IF A$ = "P" OR A$ = "p" THEN J = 2: LOCATE 15, 54: PRINT ",psia": PRINT ;
                IF A$ = "R" OR A$ = "r" THEN J = 4: LOCATE 16, 54: PRINT ",miln": PRINT ;
                IF A$ = "G" OR A$ = "g" THEN J = 6: LOCATE 17, 54: PRINT ",opn": PRINT ;
                IF A$ = "C" OR A$ = "c" THEN J = 8: LOCATE 18, 54: PRINT ",in": PRINT ;
        COLOR 16, 6
                IF A$ = "M" OR A$ = "m" THEN J = 3: LOCATE 15, 71: PRINT ",Mach": PRINT ;
                IF A$ = "N" OR A$ = "n" THEN J = 7: LOCATE 16, 71: PRINT ",rpm": PRINT ;
        COLOR 16, 2
                IF A$ = "B" OR A$ = "b" THEN J = 9: LOCATE 15, 19: PRINT ",psi": PRINT ;

                IF J = 1 THEN LU = 15: MU = 30
                IF J = 2 THEN LU = 15: MU = 48
                IF J = 3 THEN LU = 15: MU = 65
                IF J = 4 THEN LU = 16: MU = 48
                IF J = 5 THEN LU = 16: MU = 30
                IF J = 6 THEN LU = 17: MU = 48
                IF J = 7 THEN LU = 16: MU = 65
                IF J = 8 THEN LU = 18: MU = 48
                IF J = 9 THEN LU = 15: MU = 13
                IF J > 0 THEN BEEP
        GOTO 5999
```

```
5100            IF J = 1 THEN 5105
                IF J = 2 THEN 5405
                IF J = 3 THEN 5205
                IF J = 4 THEN 5405
                IF J = 5 THEN 5405
                IF J = 6 THEN 5405
                IF J = 7 THEN 5350
                IF J = 8 THEN 5405
                IF J = 9 THEN 5105

'Assemble Temperature and Liquid Back Pressure Inputs.
5105    XX = XX + 1
        D1 = 1: D2 = 2: D3 = 3: D4 = 4: D5 = 5
        GOSUB 5500
        AA = H1 * 100 + H2 * 10 + H3 + H5 / 10
        GOTO 5999

'Assemble Mach Number Input.
5205    XX = XX + 1
        D1 = 1: D2 = 3: D3 = 4: D4 = 2: D5 = 5
        GOSUB 5500
        AA = H2 / 10 + H3 / 100 + H5 / 1000
        GOTO 5999

'Assemble Fan Speed Input.
5350    XX = XX + 1
        D1 = 1: D2 = 2: D3 = 3: D4 = 5: D5 = 4
        GOSUB 5500
        AA = H1 * 1000 + H2 * 100 + H3 * 10 + H5
        GOTO 5999

'Assemble Pressure, Reynolds Number, Injection Valve, Exhaust Valve, and Chord Inputs.
5405    XX = XX + 1
        D1 = 1: D2 = 2: D3 = 4: D4 = 3: D5 = 5
        GOSUB 5500
        AA = H1 * 10 + H2 + H3 / 10 + H5 / 100
        GOTO 5999

5500            IF J = 1 THEN COLOR 0, 3
                IF J = 2 THEN COLOR 0, 5
                IF J = 3 THEN COLOR 0, 6
                IF J = 4 THEN COLOR 0, 5
                IF J = 5 THEN COLOR 0, 3
                IF J = 6 THEN COLOR 0, 5
                IF J = 7 THEN COLOR 0, 6
                IF J = 8 THEN COLOR 0, 5
                IF J = 9 THEN COLOR 0, 2
                IF XX = D1 THEN H1 = VAL(A$): LOCATE LU, MU + D1: PRINT USING "#"; H1
                IF XX = D2 THEN H2 = VAL(A$): LOCATE LU, MU + D2: PRINT USING "#"; H2
                IF XX = D3 THEN H3 = VAL(A$): LOCATE LU, MU + D3: PRINT USING "#"; H3
                IF XX = D4 THEN LOCATE LU, MU + D4: PRINT CHR$(46);
                IF XX = D5 THEN H5 = VAL(A$): LOCATE LU, MU + D5: PRINT USING "#"; H5
        RETURN
```

```
'Set Temperature Input.
5600            IF J = 1 THEN 5620 ELSE 5660
5620    ST1 = AA
                IF ST1 > MAXT THEN ST1 = MAXT
                IF ST1 < MINT THEN ST1 = MINT
                IF ST1 < SAT1 THEN ST1 = SAT1
        COLOR 14, 3
        LOCATE 3, 29: PRINT "  AUTO  ": PRINT ;
        AUTOT = 1
        LOCATE 5, 26: PRINT USING "###.#"; ST1

'Set Pressure Input.
5660            IF J = 2 THEN 5670 ELSE 5730
5670    SP = AA
                IF SP < MINP THEN SP = MINP
                IF SP > MAXP THEN SP = MAXP
        COLOR 14, 5
        LOCATE 3, 46: PRINT " AUTOP  ": PRINT ;
        LOCATE 6, 43: PRINT "      ": PRINT ;
        LOCATE 5, 43: PRINT USING "##.##"; SP
        AUTOP = 1
        AUTORE = 0

'Set Mach Number Input.
5730            IF J = 3 THEN 5740 ELSE 5750
5740    SM = AA
                IF SM < MINM THEN SM = MINM
                IF SM > MAXM THEN SM = MAXM
        COLOR 14, 6
        LOCATE 3, 64: PRINT " AUTO   ": PRINT ;
        LOCATE 6, 62: PRINT "      "
        AUTOM = 1
        LOCATE 5, 62: PRINT USING "#.###"; SM

'Set Reynolds Number Input.
5750            IF J = 4 THEN 5760 ELSE 5770
5760    SRE = AA
                IF SRE > MAXRE THEN SRE = MAXRE
                IF SRE < MINRE THEN SRE = MINRE
        COLOR 14, 5
        AUTORE = 1
        AUTOP = 1
        LOCATE 6, 43: PRINT USING "##.##"; SRE
        LOCATE 3, 46: PRINT " AUTORE ": PRINT ;

'Set Liquid Injection Valve Input.
5770            IF J = 5 THEN 5780 ELSE 5810
5780    LCMDS = AA
        COLOR 14, 3
        LOCATE 3, 29: PRINT " MANUAL ": PRINT ;
        LOCATE 5, 26: PRINT "      "
        LOCATE 6, 26: PRINT "      "
        AUTOT = 0

'Set Gas Exhaust Valve Input.
5810            IF J = 6 THEN 5820 ELSE 5860
5820    GCMDS = AA
```

```
        COLOR 14, 5
        LOCATE 3, 46: PRINT " MANUAL ": PRINT ;
        AUTOP = 0
        AUTORE = 0
        LOCATE 5, 43: PRINT "      ": PRINT ;
        LOCATE 6, 43: PRINT "      ": PRINT ;

'Set Fan Speed Input.
5860            IF J = 7 THEN 5870 ELSE 5935
5870    NCMDS = AA
                IF NCMDS > MAXN THEN NCMDS = MAXN
        COLOR 14, 6
        LOCATE 5, 62: PRINT "       "
        LOCATE 3, 64: PRINT " MANUAL ": PRINT ;
        LOCATE 6, 62: PRINT USING "####."; NCMDS
        AUTOM = 0

'Set Chord Length Input.
5935            IF J = 8 THEN 5937 ELSE 5943
5937    CHIN = AA
                IF CHIN < MINCHIN THEN CHIN = MINCHIN
                IF CHIN > MAXCHIN THEN CHIN = MAXCHIN
        COLOR 14, 5
        LOCATE 20, 48: PRINT USING "##.##"; CHIN
        CH = CHIN * .0254

'Set Liquid Back Pressure Input.
5943            IF J = 9 THEN 5944 ELSE 5950
5944    SLQSC = AA
                IF SLQSC > MAXLQSC THEN SLQSC = MAXLQSC
        COLOR 14, 2
        LOCATE 5, 13: PRINT USING "###.#"; SLQSC

5950            IF J = 0 THEN 5962
                IF J = 1 THEN COLOR 14, 3
                IF J = 2 THEN COLOR 14, 5
                IF J = 3 THEN COLOR 14, 6
                IF J = 4 THEN COLOR 14, 5
                IF J = 5 THEN COLOR 14, 3
                IF J = 6 THEN COLOR 14, 5
                IF J = 7 THEN COLOR 14, 6
                IF J = 8 THEN COLOR 14, 5
                IF J = 9 THEN COLOR 14, 2
        LOCATE LU, MU + 1: PRINT "  ": PRINT ;

5962            IF J > 0 THEN BEEP
                IF IE = 1 THEN COLOR 0, 6
                IF IE = 1 THEN BEEP: LOCATE 20, 60: PRINT "                ": PRINT ;
                IF PIE = 1 THEN LOCATE 21, 60: PRINT "            ": PIE = 0
                IF NIE = 1 THEN LOCATE 21, 60: PRINT "            ": NIE = 0
        J = 0
        XX = 0
        IE = 0

5999    RETURN
```

```
'Emergency Stop Subroutine.
6000    IE = 1
        COLOR 20, 8
        LOCATE 20, 60: PRINT " EMERGENCY STOP ": PRINT ;
        COLOR 4, 8
                IF PIE = 1 THEN LOCATE 21, 60: PRINT " Pressure Limit "
                IF NIE = 1 THEN LOCATE 21, 60: PRINT " Fan RPM Limit  "
        COLOR 4, 8
        AUTOT = 0
        AUTOP = 0
        AUTOM = 0
        AUTORE = 0
        LOCATE 3, 29: PRINT " MANUAL ": PRINT ;
        LOCATE 3, 46: PRINT " MANUAL ": PRINT ;
        LOCATE 3, 64: PRINT " MANUAL ": PRINT ;
        RETURN

'Analog to Digital Conversion Input Subroutine.
8000    OUT COMMAND.REGISTER, ZSTOP
        ZEMP = INP(ZATA.REGISTER)
        WAIT ZTATUS.REGISTER, ZOMMAND.WAIT
        OUT ZOMMAND.REGISTER, ZCLEAR
        ZDGAIN = 1
        FOR I = 1 TO 7
        ZDCHNL = I - 1
        WAIT ZTATUS.REGISTER, WRITE.WAIT, WRITE.WAIT
        WAIT ZTATUS.REGISTER, ZOMMAND.WAIT
        OUT ZOMMAND.REGISTER, ZADIN
        WAIT ZTATUS.REGISTER, WRITE.WAIT, WRITE.WAIT
        OUT ZATA.REGISTER, ZDGAIN
        WAIT ZTATUS.REGISTER, WRITE.WAIT, WRITE.WAIT
        OUT ZATA.REGISTER, ZDCHNL
        WAIT ZTATUS.REGISTER, READ.WAIT
        ZOW = INP(ZATA.REGISTER)
        WAIT ZTATUS.REGISTER, READ.WAIT
        ZIGH = INP(ZATA.REGISTER)
        ZOLT# = ZIGH * 256 + ZOW
                IF ZOLT# > 32767 THEN ZOLT# = ZOLT# - 65536!
        WAIT ZTATUS.REGISTER, ZOMMAND.WAIT
        ZTATUS = INP(ZTATUS.REGISTER)
                IF (ZTATUS AND &H80) THEN GOTO 8000
        ZOL# = 5 * ZOLT# / 32768
        E(I) = ZOL#
        NEXT I

'Sensor Failure Detection.
        E11 = E(1): IF E11 < 0 THEN E11 = 0
                IF E(1) < .43 THEN E(1) = .43: FL10 = 1
        E12 = E(2): IF E12 < 0 THEN E12 = 0
                IF E(2) < .4 THEN E(2) = .4: FL10 = 2
                IF E(3) < 0 THEN E(3) = 0!: FL10 = 3
                IF E(4) < 0! THEN E(4) = 0: FL10 = 4
                IF E(5) < .001 THEN E(5) = .001
                IF E(6) < .3 THEN E(6) = .3
                IF ABS(E(7)) > 1! THEN FL10 = 7
                IF FL10 >= 1 THEN 8340 ELSE 8380
8340    COLOR 4, 8
```

```
        LOCATE 21, 60: PRINT " Sensor Failure ": PRINT ;
                IF FL10 = 1 THEN LOCATE 22, 60: PRINT " Total Pressure  "
                IF FL10 = 2 THEN LOCATE 22, 60: PRINT " Static Pressure "
                IF FL10 = 3 THEN LOCATE 22, 60: PRINT " Gas Temperature "
                IF FL10 = 4 THEN LOCATE 22, 60: PRINT " Wall Temperature"
                IF FL10 = 7 THEN LOCATE 22, 60: PRINT " Screen Pressure "
        GOSUB 6000
        GOTO 8400
8380    COLOR 6, 6
        LOCATE 21, 60: PRINT "                    ": PRINT ;
        LOCATE 22, 60: PRINT "                    ": PRINT ;
8400    FL10 = 0
        RETURN

'Digital to Analog Conversion Output Subroutine.
9000    FOR IK = 1 TO 7
        DR1 = INT(DAC(IK) * 4095)
        HI = INT(DR1 / 16)
        DR2 = INT(DR1 - HI * 16)
                IF IK < 3 THEN 9050 ELSE 9070
9050    LO = INT(DR2 * 16) + 2 * IK - 2
        GOTO 9080
9070    LO = INT(DR2 * 16) + 2 * IK - 1
9080    BEMP = INP(&H225)
                IF BEMP = 0 THEN 9110
9100    GOTO 9080
9110    OUT &H224, LO
9120    CEMP = INP(&H225)
                IF CEMP = 16 THEN 9150
        GOTO 9120
9150    OUT &H224, HI
        NEXT IK
        RETURN

10000   END
```

Appendix B
Controller Simulator Source Code Listing

```
'0.3-m Transonic Cryogenic Tunnel Controller Simulator.
        CLS
        CLEAR

'Program Constants.
        DEL = .1
        CGV = 8
        CII = .18
        CIIN = .18 * .0245
        CLQV = 4
        IW = 39
        K = 1
        KDL = 0!
        KDP = 0!
        KDT = 0!
        KIL = .02
        KIM = 0!
        KIN = .4
        KIP = .05
        KIT = .1
        KMM = .3
        KP = .3
        KPL = .2
        KPM = 4.5
        KPN = .6
        KPP = 1!
        KPT = 1!
        KT = .04
        MXT = 40
        PLO1 = 3!
        PLQ = 3!
        PP = 1!
        PP1 = 1!
        SLQSC = 17!
        STP = 300!
        TMWL = 300!
        TMWL1 = 300!
        TT = 300!
        TT1 = 300!

'Maximum - Minimum Limits Screen Layout.
        COLOR 14, 9
        CLS
        LOCATE 3, 16: PRINT " Maximum - Minimum Limits for 0.3-m TCT Controller "

        COLOR 12, 9
        LOCATE 5, 25: PRINT "Tunnel"
        LOCATE 6, 24: PRINT "Maximun"
        LOCATE 5, 38: PRINT "Set"
        LOCATE 6, 36: PRINT "Maximun"
        LOCATE 5, 60: PRINT "Set"
        LOCATE 6, 58: PRINT "Minimum"
        LOCATE 5, 72: PRINT "Tunnel"
```

```
LOCATE 6, 71: PRINT "Minimum"

COLOR 7, 9
LOCATE 7, 3: PRINT "Temperature"
LOCATE 8, 4: PRINT "(kelvin)"
LOCATE 10, 3: PRINT "Pressure"
LOCATE 11, 4: PRINT "(psia)"
LOCATE 13, 3: PRINT "Normal Fan Speed"
LOCATE 14, 4: PRINT "(rpm)"
LOCATE 16, 3: PRINT "Mach Number"
LOCATE 19, 3: PRINT "Reynolds Number"
LOCATE 20, 4: PRINT "(Miln)"
LOCATE 22, 3: PRINT "Accept Tunnel Max./Min. Limits and Start 0.3-m TCT Controller." LOCATE
23, 3: PRINT "Accept Set Max./Min. Limits and Start 0.3-m TCT Controller."

COLOR 15, 9
LOCATE 7, 51: PRINT "T"
LOCATE 7, 49: PRINT CHR$(243)
LOCATE 7, 53: PRINT CHR$(243)
LOCATE 10, 51: PRINT "P"
LOCATE 10, 49: PRINT CHR$(243)
LOCATE 10, 53: PRINT CHR$(243)
LOCATE 13, 51: PRINT "N"
LOCATE 13, 49: PRINT CHR$(243)
LOCATE 13, 53: PRINT CHR$(243)
LOCATE 16, 51: PRINT "M"
LOCATE 16, 49: PRINT CHR$(243)
LOCATE 16, 53: PRINT CHR$(243)
LOCATE 19, 51: PRINT "R"
LOCATE 19, 49: PRINT CHR$(243)
LOCATE 19, 53: PRINT CHR$(243)

COLOR 14, 9
LOCATE 7, 3: PRINT "T"
LOCATE 10, 3: PRINT "P"
LOCATE 13, 3: PRINT "N"
LOCATE 16, 3: PRINT "M"
LOCATE 19, 3: PRINT "R"
LOCATE 22, 3: PRINT "A"
LOCATE 23, 10: PRINT "S"

'Tunnel Maximum - Minimum Limits.
MAXT1 = 340!
MINT1 = 77!
MAXP1 = 88!
MINP1 = 14.696
MAXN1 = 5600!
MINN1 = 0!
MAXM1 = .995
MINM1 = .15
MAXRE1 = 50!
MINRE1 = 1!
MAXCHIN = 15.75
MINCHIN = .394
MAXCH = .4
MINCH = .01
```

```
        MAXLQSC = 150!
        MAXSPR = MAXP1 / 14.696
        MINSPR = MINP1 / 14.696

        MAXT = MAXT1
        MINT = MINT1
        MAXP = MAXP1
        MINP = MINP1
        MAXN = MAXN1
        MINN = MINN1
        MAXM = MAXM1
        MINM = MINM1
        MAXRE = MAXRE1
        MINRE = MINRE1

        COLOR 10, 9
        LOCATE 7, 25: PRINT USING "###.#"; MAXT1
        LOCATE 7, 37: PRINT USING "###.#"; MAXT1
        LOCATE 10, 25: PRINT USING "##.##"; MAXP1
        LOCATE 10, 59: PRINT USING "##.##"; MINP1
        LOCATE 13, 25: PRINT USING "####."; MAXN1
        LOCATE 13, 60: PRINT USING "####."; MINN1
        LOCATE 16, 25: PRINT USING "#.###"; MAXM1
        LOCATE 16, 60: PRINT USING "#.###"; MINM1
        LOCATE 19, 25: PRINT USING "##.##"; MAXRE1
        LOCATE 19, 60: PRINT USING "##.##"; MINRE1
        LOCATE 7, 69: PRINT "Saturation"
        LOCATE 8, 69: PRINT "Temperature"
        LOCATE 10, 71: PRINT "Ambient"
        LOCATE 11, 71: PRINT "Pressure"
        LOCATE 13, 73: PRINT USING "####."; MINN1
        LOCATE 16, 73: PRINT USING "#.###"; MINM1
        LOCATE 19, 73: PRINT USING "##.##"; MINRE1

'Maximum - Minimum Limits Inputs.
5       A$ = INKEY$
                IF A$ = "T" OR A$ = "t" THEN GOSUB 10
                IF A$ = "P" OR A$ = "p" THEN GOSUB 100
                IF A$ = "N" OR A$ = "n" THEN GOSUB 200
                IF A$ = "M" OR A$ = "m" THEN GOSUB 300
                IF A$ = "R" OR A$ = "r" THEN GOSUB 400
                IF A$ = "A" OR A$ = "a" THEN 500
                IF A$ = "S" OR A$ = "s" THEN 600
        GOTO 5

'Temperature Minimum Limit Subroutine.
10      COLOR 15, 9
        BEEP
        LR = 7
        LC = 58
        LOCATE LR, LC + 1: PRINT "      ,K"
        XX = 0
20      B$ = INKEY$
                IF B$ = "D" OR B$ = "d" THEN 30 ELSE 40
30      LOCATE LR, LC + 1: PRINT "          "
        RETURN
40              IF B$ = "0" OR B$ = "1" OR B$ = "2" OR B$ = "3" OR B$ = "4" OR B$ = "5" OR B$ =
```

```
            "6" OR B$ = "7" OR B$ = "8" OR B$ = "9" OR B$ = "." THEN 50 ELSE 20
50          XX = XX + 1
            IF XX = 1 THEN H1 = VAL(B$): LOCATE LR, LC + 1: PRINT USING "#"; H1
            IF XX = 2 THEN H2 = VAL(B$): LOCATE LR, LC + 2: PRINT USING "#"; H2
            IF XX = 3 THEN H3 = VAL(B$): LOCATE LR, LC + 3: PRINT USING "#"; H3
            IF XX = 4 THEN LOCATE LR, LC + 4: PRINT CHR$(46)
            IF XX = 5 THEN H5 = VAL(B$): LOCATE LR, LC + 5: PRINT USING "#"; H5
            IF XX = 5 THEN 60 ELSE 20
60      AA = H1 * 100 + H2 * 10 + H3 + H5 / 10
        MINT = AA
            IF MINT >= MAXT1 THEN MINT = MAXT1
            IF MINT <= MINT1 THEN MINT = MINT1
70      C$ = INKEY$
            IF C$ = CHR$(13) THEN 80 ELSE 70
80      BEEP
        COLOR 14, 9
        LOCATE LR, LC + 1: PRINT USING "###.#  "; MINT
        RETURN

'Pressure Maximum Limit Subroutine.
100     COLOR 15, 9
        BEEP
        LC = 36
        LR = 10
        LOCATE LR, LC + 1: PRINT "      ,psia"
        XX = 0
110     B$ = INKEY$
            IF B$ = "D" OR B$ = "d" THEN 120 ELSE 130
120     LOCATE LR, LC + 1: PRINT "    "
        RETURN
130             IF B$ = "0" OR B$ = "1" OR B$ = "2" OR B$ = "3" OR B$ = "4" OR B$ = "5" OR B$ =
                "6" OR B$ = "7" OR B$ = "8" OR B$ = "9" OR B$ = "." THEN 140 ELSE 110
140     XX = XX + 1
            IF XX = 1 THEN H1 = VAL(B$): LOCATE LR, LC + 1: PRINT USING "#"; H1
            IF XX = 2 THEN H2 = VAL(B$): LOCATE LR, LC + 2: PRINT USING "#"; H2
            IF XX = 4 THEN H3 = VAL(B$): LOCATE LR, LC + 4: PRINT USING "#"; H3
            IF XX = 3 THEN LOCATE LR, LC + 3: PRINT CHR$(46)
            IF XX = 5 THEN H5 = VAL(B$): LOCATE LR, LC + 5: PRINT USING "#"; H5
            IF XX = 5 THEN 150 ELSE 110
150     AA = H1 * 10 + H2 + H3 / 10 + H5 / 100
        MAXP = AA
            IF MAXP <= MINP1 THEN 160 ELSE 170
160     MAXP = MINP1
        COLOR 20, 8: LOCATE LR + 1, 36: PRINT " Warning!  Set Max. = Tunnel Min. "
        FLO = 1
        GOTO 180
170     COLOR 14, 9: LOCATE LR + 1, 36: PRINT "                                    "
        FLO = 0
180             IF MAXP > MAXP1 THEN MAXP = MAXP1
190     C$ = INKEY$
            IF C$ = CHR$(13) THEN 195 ELSE 190
195     BEEP
        COLOR 14, 9
        LOCATE LR, LC + 1: PRINT USING "##.##      "; MAXP
        RETURN
```

```
'Normal Fan Speed Maximum Limit Subroutine.
200     COLOR 15, 9
        BEEP
        LC = 36
        LR = 13
        LOCATE LR, LC + 1: PRINT "      ,rpm"
        XX = 0
210     B$ = INKEY$
            IF B$ = "D" OR B$ = "d" THEN 220 ELSE 230
220     LOCATE LR, LC + 1: PRINT "   "
        RETURN
230         IF B$ = "0" OR B$ = "1" OR B$ = "2" OR B$ = "3" OR B$ = "4" OR B$ = "5" OR B$ =
            "6" OR B$ = "7" OR B$ = "8" OR B$ = "9" OR B$ = "." THEN 240 ELSE 210
240     XX = XX + 1
            IF XX = 1 THEN H1 = VAL(B$): LOCATE LR, LC + 1: PRINT USING "#"; H1
            IF XX = 2 THEN H2 = VAL(B$): LOCATE LR, LC + 2: PRINT USING "#"; H2
            IF XX = 3 THEN H3 = VAL(B$): LOCATE LR, LC + 3: PRINT USING "#"; H3
            IF XX = 5 THEN LOCATE LR, LC + 5: PRINT CHR$(46)
            IF XX = 4 THEN H5 = VAL(B$): LOCATE LR, LC + 4: PRINT USING "#"; H5
            IF XX = 5 THEN 250 ELSE 210
250     AA = H1 * 1000 + H2 * 100 + H3 * 10 + H5
        MAXN = AA
            IF MAXN <= MINN1 THEN 260 ELSE 270
260     MAXN = MINN1
        COLOR 20, 8: LOCATE LR + 1, 36: PRINT " Warning!  Set Max. = Tunnel Min. "
        FL0 = 1
        GOTO 280
270     COLOR 14, 9: LOCATE LR + 1, 36: PRINT "                                    "
        FL0 = 0
280         IF MAXN > MAXN1 THEN MAXN = MAXN1
290     C$ = INKEY$
            IF C$ = CHR$(13) THEN 295 ELSE 290
295     BEEP
        COLOR 14, 9
        LOCATE LR, LC + 1: PRINT USING "####.      "; MAXN
        RETURN

'Mach Number Maximum Limit Subroutine.
300     COLOR 15, 9
        BEEP
        LC = 36
        LR = 16
        LOCATE LR, LC + 1: PRINT "      ,Mach"
        XX = 0
310     B$ = INKEY$
            IF B$ = "D" OR B$ = "d" THEN 320 ELSE 330
320     LOCATE LR, LC + 1: PRINT "   "
        RETURN
330         IF B$ = "0" OR B$ = "1" OR B$ = "2" OR B$ = "3" OR B$ = "4" OR B$ = "5" OR B$ =
            "6" OR B$ = "7" OR B$ = "8" OR B$ = "9" OR B$ = "." THEN 340 ELSE 310
340     XX = XX + 1
            IF XX = 1 THEN H1 = VAL(B$): LOCATE LR, LC + 1: PRINT USING "#"; H1
            IF XX = 3 THEN H2 = VAL(B$): LOCATE LR, LC + 3: PRINT USING "#"; H2
            IF XX = 4 THEN H3 = VAL(B$): LOCATE LR, LC + 4: PRINT USING "#"; H3
            IF XX = 2 THEN LOCATE LR, LC + 2: PRINT CHR$(46)
            IF XX = 5 THEN H5 = VAL(B$): LOCATE LR, LC + 5: PRINT USING "#"; H5
            IF XX = 5 THEN 350 ELSE 310
```

```
350     AA = H1 + H2 / 10 + H3 / 100 + H5 / 1000
        MAXM = AA
                IF MAXM <= MINM1 THEN 360 ELSE 370
360     MAXM = MINM1
        COLOR 20, 8: LOCATE LR + 1, 36: PRINT " Warning!  Set Max. = Tunnel Min. "
        FLO = 1
        GOTO 380
370     COLOR 14, 9: LOCATE LR + 1, 36: PRINT "                                    "
        FLO = 0
380             IF MAXM > MAXM1 THEN MAXM = MAXM1
390     C$ = INKEY$
                IF C$ = CHR$(13) THEN 395 ELSE 390
395     BEEP
        COLOR 14, 9
        LOCATE LR, LC + 1: PRINT USING "#.###      "; MAXM
        RETURN

'Reynolds Number Maximum Limit Subroutine.
400     COLOR 15, 9
        BEEP
        LC = 36
        LR = 19
        LOCATE LR, LC + 1: PRINT "      ,Miln"
        XX = 0
410     B$ = INKEY$
                IF B$ = "D" OR B$ = "d" THEN 420 ELSE 430
420     LOCATE LR, LC + 1: PRINT "    "
        RETURN
430             IF B$ = "0" OR B$ = "1" OR B$ = "2" OR B$ = "3" OR B$ = "4" OR B$ = "5" OR B$ =
                "6" OR B$ = "7" OR B$ = "8" OR B$ = "9" OR B$ = "." THEN 440 ELSE 410
440     XX = XX + 1
                IF XX = 1 THEN H1 = VAL(B$): LOCATE LR, LC + 1: PRINT USING "#"; H1
                IF XX = 2 THEN H2 = VAL(B$): LOCATE LR, LC + 2: PRINT USING "#"; H2
                IF XX = 4 THEN H3 = VAL(B$): LOCATE LR, LC + 4: PRINT USING "#"; H3
                IF XX = 3 THEN LOCATE LR, LC + 3: PRINT CHR$(46)
                IF XX = 5 THEN H5 = VAL(B$): LOCATE LR, LC + 5: PRINT USING "#"; H5
                IF XX = 5 THEN 450 ELSE 410
450     AA = H1 * 10 + H2 + H3 / 10 + H5 / 100
        MAXRE = AA
                IF MAXRE <= MINRE1 THEN 460 ELSE 470
460     MAXRE = MINRE1
        COLOR 20, 8: LOCATE LR + 1, 36: PRINT " Warning!  Set Max. = Tunnel Min. "
        FLO = 1
        GOTO 480
470     COLOR 14, 9: LOCATE LR + 1, 36: PRINT "                                    "
        FLO = 0
480             IF MAXRE > MAXRE1 THEN MAXRE = MAXRE1
490     C$ = INKEY$
                IF C$ = CHR$(13) THEN 495 ELSE 490
495     BEEP
        COLOR 14, 9
        LOCATE LR, LC + 1: PRINT USING "##.##      "; MAXRE
        RETURN

'Starts the Controller with the Tunnel Maximum - Minimum Limits.
500     XX = 0
        GOTO 800
```

```
'Starts the Controller with the Set Maximum - Minimum Limits.
600             IF FLO = 1 THEN GOTO 5
        XX = 0
        MAXSPR = MAXP / 14.696
        MINSPR = MINP / 14.696
        GOTO 800

'Controller Simulation.
'Controller Screen Layout.
800     COLOR 0, 0
        CLS

        COLOR 15, 4
        LOCATE 1, 26: PRINT " 0.3-m TUNNEL T-P/R-M CONTROLLER "

        FOR I = 2 TO 25
        COLOR 15, 1
        LOCATE I, 2: PRINT STRING$(10, 0);
                IF I = 4 THEN LOCATE 4, 2: PRINT STRING$(10, 205);
                IF I = 14 THEN LOCATE 14, 2: PRINT STRING$(10, 205);
                IF I = 19 THEN LOCATE 19, 2: PRINT STRING$(10, 205);
                IF I = 23 THEN LOCATE 23, 2: PRINT STRING$(18, 205);
        COLOR 15, 2
        LOCATE I, 12: PRINT STRING$(12, 0);
                IF I = 4 THEN LOCATE 4, 12: PRINT STRING$(12, 205);
                IF I = 14 THEN LOCATE 14, 12: PRINT STRING$(12, 205);
                IF I = 19 THEN LOCATE 19, 12: PRINT STRING$(12, 205);
                IF I = 23 THEN LOCATE 23, 12: PRINT STRING$(18, 205);
        COLOR 15, 3
        LOCATE I, 24: PRINT STRING$(17, 0);
                IF I = 4 THEN LOCATE 4, 24: PRINT STRING$(17, 205);
                IF I = 14 THEN LOCATE 14, 24: PRINT STRING$(17, 205);
                IF I = 19 THEN LOCATE 19, 24: PRINT STRING$(17, 205);
                IF I = 23 THEN LOCATE 23, 24: PRINT STRING$(18, 205);
        COLOR 15, 5
        LOCATE I, 41: PRINT STRING$(18, 0);
                IF I = 4 THEN LOCATE 4, 41: PRINT STRING$(18, 205);
                IF I = 14 THEN LOCATE 14, 41: PRINT STRING$(18, 205);
                IF I = 19 THEN LOCATE 19, 41: PRINT STRING$(18, 205);
                IF I = 23 THEN LOCATE 23, 41: PRINT STRING$(18, 205);
        COLOR 15, 6
        LOCATE I, 59: PRINT STRING$(18, 0);
                IF I = 4 THEN LOCATE 4, 59: PRINT STRING$(18, 205);
                IF I = 14 THEN LOCATE 14, 59: PRINT STRING$(18, 205);
                IF I = 19 THEN LOCATE 19, 59: PRINT STRING$(18, 205):
                IF I = 23 THEN LOCATE 23, 59: PRINT STRING$(18, 205);
        NEXT I

        COLOR 15, 2
        LOCATE 2, 14: PRINT " LN PUMP ": PRINT ;
        LOCATE 3, 14: PRINT "  AUTO ": PRINT ;
        LOCATE 5, 18: PRINT ",psia": PRINT ;
        LOCATE 9, 18: PRINT ",psia": PRINT ;
        LOCATE 12, 18: PRINT ",%opn": PRINT ;
        LOCATE 15, 12: PRINT " =": PRINT ;
        LOCATE 5, 13: PRINT USING "###.#"; SLQSC
```

```
COLOR 14, 2
LOCATE 15, 12: PRINT "B": PRINT ;

COLOR 15, 3
LOCATE 2, 28: PRINT "TEMP LOOP": PRINT ;
LOCATE 3, 29: PRINT " MANUAL ": PRINT ;
LOCATE 5, 31: PRINT ",K(Final)": PRINT ;
LOCATE 6, 31: PRINT ",K(Use)": PRINT ;
LOCATE 9, 31: PRINT ",K-GN2": PRINT ;
LOCATE 10, 31: PRINT ",K-WALL": PRINT ;
LOCATE 12, 31: PRINT ",%opn": PRINT ;
LOCATE 15, 25: PRINT " emp=": PRINT ;
LOCATE 16, 25: PRINT "A Q%=": PRINT ;
LOCATE 20, 25: PRINT "GRAD=       ,K/mt": PRINT ;
LOCATE 21, 25: PRINT " SAT=       ,K": PRINT ;
LOCATE 24, 32: PRINT "T";
LOCATE 24, 30: PRINT CHR$(243);
LOCATE 24, 34: PRINT CHR$(243);
LOCATE 24, 26: PRINT USING "###"; MINT;
LOCATE 24, 36: PRINT USING "###"; MAXT;

COLOR 14, 3
LOCATE 15, 25: PRINT "T": PRINT ;
LOCATE 16, 26: PRINT "L": PRINT ;

COLOR 15, 5
LOCATE 2, 45: PRINT "Pt/Rc LOOP": PRINT ;
LOCATE 3, 46: PRINT " MANUAL ": PRINT ;
LOCATE 5, 48: PRINT ",Psia": PRINT ;
LOCATE 6, 48: PRINT ",Miln": PRINT ;
LOCATE 20, 42: PRINT "CHORD= 7.09,in": PRINT ;
LOCATE 9, 48: PRINT ",Psia": PRINT ;
LOCATE 21, 42: PRINT "P st=       ,psia": PRINT ;
LOCATE 22, 42: PRINT "Del P=       ,psi": PRINT ;
LOCATE 10, 48: PRINT ",Miln": PRINT ;
LOCATE 12, 48: PRINT ",%opn V1": PRINT ;
LOCATE 13, 48: PRINT ",%opn V2": PRINT ;
LOCATE 15, 43: PRINT " res=": PRINT ;
LOCATE 16, 43: PRINT " yno=": PRINT ;
LOCATE 17, 43: PRINT "A v%=": PRINT ;
LOCATE 18, 43: PRINT " hrd=": PRINT ;
LOCATE 24, 50: PRINT "P";
LOCATE 24, 48: PRINT CHR$(243);
LOCATE 24, 52: PRINT CHR$(243);
LOCATE 24, 43: PRINT USING "##.#"; MINP;
LOCATE 24, 54: PRINT USING "##.#"; MAXP;
LOCATE 25, 50: PRINT "R";
LOCATE 25, 48: PRINT CHR$(243);
LOCATE 25, 52: PRINT CHR$(243);
LOCATE 25, 43: PRINT USING "##.#"; MINRE;
LOCATE 25, 54: PRINT USING "##.#"; MAXRE;

COLOR 14, 5
LOCATE 15, 43: PRINT "P": PRINT ;
LOCATE 16, 43: PRINT "R": PRINT ;
LOCATE 17, 44: PRINT "G": PRINT ;
```

```
        LOCATE 18, 43: PRINT "C": PRINT ;

        COLOR 15, 6
        LOCATE 2, 62: PRINT "RPM/MACH LOOP": PRINT
        LOCATE 3, 64: PRINT " MANUAL ": PRINT ;
        LOCATE 5, 67: PRINT ",Mach": PRINT ;
        LOCATE 6, 67: PRINT ",RPM": PRINT ;
        LOCATE 9, 67: PRINT ",Mach": PRINT ;
        LOCATE 10, 67: PRINT ",RPM": PRINT ;
        LOCATE 12, 67: PRINT ",% RhsT": PRINT ;
        LOCATE 15, 61: PRINT " ach=": PRINT ;
        LOCATE 16, 61: PRINT " rpm=": PRINT ;
        LOCATE 24, 68: PRINT "N";
        LOCATE 24, 66: PRINT CHR$(243);
        LOCATE 24, 70: PRINT CHR$(243);
        LOCATE 24, 61: PRINT USING "####"; MINN;
        LOCATE 24, 72: PRINT USING "####"; MAXN;
        LOCATE 25, 68: PRINT "M";
        LOCATE 25, 66: PRINT CHR$(243);
        LOCATE 25, 70: PRINT CHR$(243);
        LOCATE 25, 61: PRINT USING ".###"; MINM;
        LOCATE 25, 72: PRINT USING ".###"; MAXM;

        COLOR 14, 6
        LOCATE 15, 61: PRINT "M": PRINT ;
        LOCATE 16, 61: PRINT "N": PRINT ;

        COLOR 15, 1
        LOCATE 5, 3: PRINT "SET POINT": PRINT ;
        LOCATE 9, 3: PRINT "PROCESS": PRINT ;
        LOCATE 12, 3: PRINT "COMMAND": PRINT ;
        LOCATE 15, 3: PRINT "INPUTS": PRINT ;
        LOCATE 20, 3: PRINT "STATUS": PRINT ;
        LOCATE 16, 3: PRINT " elete": PRINT ;
        LOCATE 24, 3: PRINT "LIMITS";

        COLOR 14, 1
        LOCATE 16, 3: PRINT "D": PRINT ;

'Simulation Controller Program.
1000    CRPM = FRPM
                IF FRPM < 0 THEN FRPM = 0
                IF CRPM < 50 THEN CRPM = 50
        CRM = 2.77 - CRPM / 157 / SQR(TT) * PP ^ .035
                IF CRM < .001 THEN CRM = .001
        M = 1.67 - SQR(CRM)
                IF M > .999 THEN M = .999
        PS = PP / (1 + .2 * M ^ 2) ^ 3.5
        PPUSCS = PP * 14.696
        PSUSCS = PS * 14.696
        PLQUSCS = PLQ * 14.696

        GOSUB 5000

        MF = (1 + .2 * M * M)
        KRE = 63714 * CH * M / TT ^ 1.4 / (MF) ^ 2.1
        RE = KRE * PP
```

```
        SAT = 50 + 27.34 * PS ^ .296
        SAT1 = SAT * MF
        LDPQ = (PLQ - PP)
                IF LDPQ < .5 THEN LDPQ = .5
        LF = CLQV * .8676 * SQR(LDPQ)
        DRPM = FRPM
                IF DRPM < 100 THEN DRPM = 100
        KTGS = DRPM * SQR(PP) * KT / 3! / TT / LF
        TTMP = TT
                IF TTMP < 80 THEN TTMP = 80
        FKW = 100 * PP * (FRPM / 1000) ^ 2.26 / SQR(TTMP)
        FB = FKW / (121 + TT) / LF
        SPR = SP / 14.696
        SPR1 = SRE / KRE
                IF SPR1 > MAXSPR THEN SPR1 = MAXSPR
                IF SPR1 < MINSPR THEN SPR1 = MINSPR
                IF AUTORE = 1 THEN SPR = SPR1
        SPRU = SPR * 14.696
        COLOR 14, 5
                IF AUTORE = 1 THEN LOCATE 5, 43: PRINT USING "##.##"; SPRU

'Temperature Control Loop.
                IF IE = 1 THEN 2950
                IF (TMWL - TT) > 1.5 * MXT THEN 2950
                IF FRPM < 580 THEN 2950
        ST = ST1
                IF ST > TT THEN 2530 ELSE 2525
2525            IF ST < (STP - .04) THEN ST = (STP - .04)
2530            IF AUTOT = 1 THEN 2540 ELSE 2850
2540            IF AUTOP = 0 THEN 2610
                IF ABS(PP - SPR) < .15 THEN 2560 ELSE 2570
2560            IF ABS(TT - ST) < 2! THEN 2610 ELSE 2570
2570            IF (PP / TT) > .95 * (SPR / ST) THEN 2610 ELSE 2580
2580            IF TT > ST THEN 2610 ELSE 2590
2590    ST = TT - .02
2610            IF (TMWL - ST - MXT) > 0 THEN 2620 ELSE 2650
2620    ST = TMWL - MXT
        FB = 0
2650            IF ST < SAT1 THEN ST = SAT1
        ET = TT - ST
                IF ABS(TT - ST1) < .3 THEN FL1 = 1 ELSE FL1 = 0
                IF ABS(TMWL - TT) < 24 THEN FL2 = 1 ELSE FL2 = 0
                IF ET < -5 THEN FF = 0
                IF ET > 0 THEN FF = 1
        FBF = FB * FF * .8
        RIT = RITM1 + KIT * KTGS * DEL * ET
                IF RIT < -FBF THEN RIT = -FBF
                IF RIT > (1 - FBF) THEN RIT = (1 - FBF)
                IF (TMWL - TT) > MXT / 2 THEN 2750 ELSE 2780
2750    LMT = 1 - (TMWL - TT - MXT / 2) * 2 / MXT
                IF LMT < 1! / LF THEN LMT = 1! / LF
                IF RIT > LMT THEN RIT = LMT
2780    ALQ = KTGS * (KPT * ET + KDT * (ET - 2 * ETM1 + ETM2) / 2 / DEL) + RIT + FBF
                IF ALQ < 0 THEN ALQ = 0
                IF ALQ > 1 THEN ALQ = 1
                IF (TMWL - TT) > MXT / 2 THEN 2820 ELSE 2830
2820            IF ALQ > LMT THEN ALQ = LMT
```

```
2830    LCMDS = ALQ * 100
        GOTO 2970
2850            IF LCMDS > 100 THEN LCMDS = 100
        ALQ = LCMDS / 100
                IF (TMWL - TT) > MXT / 2 THEN 2880 ELSE 2910
2880    LMT = 1 - (TMWL - TT - MXT / 2) * 2 / MXT
                IF LMT < 1! / LF THEN LMT = 1! / LF
                IF ALQ > LMT THEN ALQ = LMT
2910    RIT = ALQ
        GOTO 2970
2950    ALQ = 0
        LCMDS = 0
2970            IF ALQ > ALQP + .01 THEN ALQ = ALQP + .01
        DAC(1) = ALQ
        DAC(2) = ALQ

'Pressure - Reynolds Number Control Loop.
        KPGS = 750 * KP / PP / SQR(TT)
        GF1 = 2.725 * CGV * PP / SQR(TT)
                IF PPUSCS > MAXP1 THEN PIE = 1: GOSUB 6000
                IF IE = 1 THEN 3220
                IF ABS(RE - SRE) < .05 THEN FL7 = 1 ELSE FL7 = 0
                IF AUTOP = 1 THEN 3040 ELSE 3180
3040    EP = PP - SPR
                IF ABS(EP) < .005 THEN FL3 = 1 ELSE FL3 = 0
        RIP = RIPM1 + EP * KPGS * KIP * DEL
                IF RIP < 0 THEN RIP = 0
                IF RIP > 1! THEN RIP = 1!
        AGV1 = KPGS * (KPP * EP + KDP * (EP - 2 * EPM1 + EPM2) / 2 / DEL) + RIP
                IF AGV1 < 0 THEN AGV1 = 0
                IF AGV1 > 1! THEN AGV1 = 1!
                IF AGV1 > .9 THEN AGV2 = AGV2 + .01
                IF AGV1 < .7 THEN AGV2 = AGV2 - .01
                IF AGV2 < 0 THEN AGV2 = 0
                IF AGV2 > 1! THEN AGV2 = 1!
        GCMDS = AGV1 * 100
        GOTO 3240
3180            IF GCMDS > 100 THEN GCMDS = 100
        RIP = GCMDS / 100
        AGV1 = GCMDS / 100
                IF FRPM < 300 THEN LCMDS = 0
        GOTO 3240
3220    AGV1 = 1!
        GCMDS = 100
3240            IF AGV1 < AGV1P - .05 THEN AGV1 = AGV1P - .05
                IF AGV1 > AGV1P + .05 THEN AGV1 = AGV1P + .05
        DAC(4) = AGV1 * .8 + .2
        DAC(5) = AGV2 * .8 + .2

'Fan Speed - Mach Number Control Loop.
                IF IE = 1 THEN 3405 ELSE 3410
3405    NCMDS1 = 0
        GOTO 3550
3410    NCMDS1 = NCMDS
                IF FRPM > MAXN1 THEN NIE = 1: GOSUB 6000
                IF AUTOM = 1 THEN 3422 ELSE 3550
3422            IF FRPM < 580 THEN AUTOM = 0: GOTO 3435
```

```
        FKM = 1 - .5 * M
              IF FKM < .71 THEN FKM = .71
        KMC = 520 * SQR(TT) * FKM / PP ^ .035
        KMGS = KMC * KMM
              IF ALQ > .99 THEN 3432 ELSE 3440
3432    AUTOM = 0
        NCMDS = FRPM - 500
              IF NCMDS < 0 THEN NCMDS = 0: BEEP
3435    COLOR 14, 6
        LOCATE 3, 64: PRINT " MANUAL ": PRINT
        LOCATE 6, 62: PRINT USING "####."; NCMDS
        GOTO 3550
3440    EM = SM - M
              IF ABS(EM) < .002 THEN FLA = 1 ELSE FLA = 0
        RIM = RIMM1 + KIM * EM * DEL * KMGS
              IF RIM < -100 THEN RIM = -100
              IF RIM > 100 THEN RIM = 100
        NCMDS1 = KPM * EM * KMGS + RIM + FRPM
              IF NCMDS1 > 5450 THEN NCMDS1 = 5450
              IF NCMDS1 > MAXN THEN NCMDS1 = MAXN
3550    EN = NCMDS1 - FRPM
           IF ABS(EN)<10 THEN FLN=1 ELSE FLN=0
              IF ABS(EN) > 100 THEN 3560 ELSE 3570
3560    KEN = 5
        GOTO 3575
3570    KEN = SQR(TT) / 20
3575    RIN = RINM1 + EN * DEL * KIN
              IF RIN > RINM1 + KEN THEN RIN = RINM1 + KEN
              IF RIN < RINM1 - KEN THEN RIN = RINM1 - KEN
              IF RIN < 0 THEN RIN = 0
              IF RIN > 6000 THEN RIN = 6000
        SRPM = EN * KPN + RIN
              IF SRPM > SRPM1 + 5 THEN SRPM = SRPM1 + 5
              IF SRPM < SRPM1 - 5 THEN SRPM = SRPM1 - 5
        SNRPM = SRPM / 7500
              IF SNRPM > 1! THEN SNRPM = 1!
              IF SNRPM < 0! THEN SNRPM = 0!
        DAC(6) = SNRPM

'Fan Speed Band Warning.
              IF FRPM < 3651 THEN 3790 ELSE 3830
3790          IF FRPM > 3549 THEN 3800 ELSE 3830
3800    COLOR 20, 8
        LOCATE 13, 61: PRINT " SPEED BAND ": PRINT
        BEEP
        GOTO 3900
3830    COLOR 14, 6
        LOCATE 13, 61: PRINT "                ": PRINT

'Temperature Gradient Calculation.
3900    IW = IW + 1
              IF IW = 40 THEN 3920 ELSE 3980
3920          IF WLG > 8 THEN WLG = 0
              IF WLG < -8 THEN WLG = 0
        COLOR 14, 3
        LOCATE 20, 30: PRINT USING "###.#"; 12 * WLG
        WLG = 0
```

```
        IW = 0
3980    WLG = WLG + TMWL - TMWL1

'Liquid Back Pressure Control Loop.
        SLQ = SLQSC / 14.696
                IF FRPM < 600 THEN SLQ = 9!
                IF FRPM < 400 THEN SLQ = .1
                IF SLQ > 10.2 THEN SLQ = 10.2
        SLQSC = SLQ * 14.696
        ELP = PLQ - SLQ
                IF ABS(ELP) < .4 THEN FL5 = 1 ELSE FL5 = 0
                IF ELP > .15 THEN ELP = .15
                IF ELP < -.15 THEN ELP = -.15
        RIL = RILM1 + ELP * KIL * DEL
                IF RIL < 0 THEN RIL = 0
                IF RIL > 1! THEN RIL = 1!
        ALN = KPL * ELP + KDL * (ELP - ELPM1) / DEL + RIL
                IF ALN < 0 THEN ALN = 0
                IF ALN > 1! THEN ALN = 1!
        DAC(3) = ALN
        DAC(7) = 1!


'Screen Pressure Drop Warning.
        DLPC = .47 * PP * M * M / (MF) ^ 6 + .02
                IF DLP > DLPC THEN FL6 = 1 ELSE FL6 = 0
                IF M < .25 THEN FL6 = 0
                IF FL6 = 1 THEN BEEP

'Controller Screen Update.
        JD = JD + 1
                IF JD = 3 THEN 4160 ELSE 4700
4160    COLOR 14, 3
        LOCATE 21, 30: PRINT USING "###.#"; SAT1
                IF (FL1 * FL2 * AUTOT) = 1 THEN 4190 ELSE 4200
4190    COLOR 14, 0
4200    LOCATE 8, 26: PRINT STRING$(6, 0)
        COLOR 14, 5
        LOCATE 21, 47: PRINT USING "###.##"; PSUSCS
                IF FL6 = 1 THEN COLOR 30, 5
        LOCATE 22, 48: PRINT USING "##.###"; DLP
                IF AUTORE = 0 THEN 4240
                IF (FL7 * AUTORE) = 1 THEN 4250 ELSE 4260
4240            IF (FL3 * AUTOP) = 1 THEN 4250 ELSE 4260
4250    COLOR 14, 0
4260    LOCATE 8, 43: PRINT STRING$(6, 0)
        COLOR 14, 2
        LOCATE 5, 13: PRINT USING "###.#"; SLQSC
        LOCATE 9, 13: PRINT USING "###.#"; PLQUSCS
        LOCATE 12, 13: PRINT USING "###.#"; (1 - ALN) * 100
                IF FL5 = 1 THEN 4320 ELSE 4330
4320    COLOR 14, 0
4330    LOCATE 8, 13: PRINT STRING$(6, 0)
        COLOR 14, 6
                IF (FL4 * AUTOM) = 1 THEN 4360 ELSE 4370
4360    COLOR 14, 0
4370    LOCATE 8, 62: PRINT STRING$(6, 0)
```

```
    IF (FLN-AUTOM)>0 THEN 4374 ELSE 4376
4374 COLOR 14,0
4376 LOCATE 11,62:PRINT STRIN$(6,000)
        COLOR 14, 3
                IF AUTOT = 1 THEN LOCATE 6, 26: PRINT USING "###.#"; ST
        LOCATE 9, 26: PRINT USING "###.#"; TT
        LOCATE 10, 26: PRINT USING "###.#"; TMWL
        LQ = ALQ * 100
        LOCATE 12, 26: PRINT USING "###.#"; LQ
        COLOR 14, 5
        LOCATE 9, 42: PRINT USING "###.##"; PPUSCS
        GV1 = AGV1 * 100
        LOCATE 12, 43: PRINT USING "###.#"; GV1
        GV2 = AGV2 * 100
        LOCATE 13, 43: PRINT USING "###.#"; GV2
        LOCATE 10, 42: PRINT USING "###.##"; RE
        COLOR 14, 6
        LOCATE 9, 62: PRINT USING "#.###"; M
        LOCATE 10, 62: PRINT USING "####."; FRPM
        LOCATE 12, 62: PRINT USING "##.#"; SNRPM * 100
        JD = 0

4700    GOSUB 7000

'Program Timing Counter.
        K = K + 1
                IF K = 600 THEN BEEP ELSE 4800
        K = 0

'Setting Previous Cycle Values.
4800    AGV1P = AGV1
        ALQP = ALQ
        ELPM1 = ELP
        EPM2 = EPM1
        EPM1 = EP
        ETM2 = ETM1
        ETM1 = ET
        RILM1 = RIL
        RIMM1 = RIM
        RINM1 = RIN
        RIPM1 = RIP
        RITM1 = RIT
        SRPM1 = SRPM
        STP = ST
        GOTO 1000

'Controller Input Subroutine.
5000    A$ = INKEY$
                IF A$ = "" THEN 5999
                IF A$ = "Q" OR A$ = "q" THEN 7500
                IF A$ = "E" OR A$ = "e" THEN 6000
                IF A$ = "D" OR A$ = "d" THEN 5950
                IF XX > 5 THEN XX = 5
                IF A$ = CHR$(13) THEN 5008 ELSE 5009
5008            IF XX = 5 THEN 5600 ELSE 5999
5009            IF J > 0 THEN 5010 ELSE 5014
5010            IF A$ = "0" OR A$ = "1" OR A$ = "2" OR A$ = "3" OR A$ = "4" OR A$ = "5" OR A$
```

```
            = "6" OR A$ = "7" OR A$ = "8" OR A$ = "9" OR A$ = "." THEN 5100 ELSE 5999
5014    COLOR 16, 3
            IF A$ = "T" OR A$ = "t" THEN J = 1: LOCATE 15, 36: PRINT ",K": PRINT ;
            IF A$ = "L" OR A$ = "l" THEN J = 5: LOCATE 16, 36: PRINT ",opn": PRINT ;
        COLOR 16, 5
            IF A$ = "P" OR A$ = "p" THEN J = 2: LOCATE 15, 54: PRINT ",psia": PRINT;
            IF A$ = "R" OR A$ = "r" THEN J = 4: LOCATE 16, 54: PRINT ",miln": PRINT;
            IF A$ = "G" OR A$ = "g" THEN J = 6: LOCATE 17, 54: PRINT ",opn": PRINT;
            IF A$ = "C" OR A$ = "c" THEN J = 8: LOCATE 18, 54: PRINT ",in": PRINT ;
        COLOR 16, 6
            IF A$ = "M" OR A$ = "m" THEN J = 3: LOCATE 15, 71: PRINT ",Mach": PRINT;
            IF A$ = "N" OR A$ = "n" THEN J = 7: LOCATE 16, 71: PRINT ",rpm": PRINT;
        COLOR 16, 2
            IF A$ = "B" OR A$ = "b" THEN J = 9: LOCATE 15, 19: PRINT ",psi": PRINT ;

            IF J = 1 THEN LU = 15: MU = 30
            IF J = 2 THEN LU = 15: MU = 48
            IF J = 3 THEN LU = 15: MU = 65
            IF J = 4 THEN LU = 16: MU = 48
            IF J = 5 THEN LU = 16: MU = 30
            IF J = 6 THEN LU = 17: MU = 48
            IF J = 7 THEN LU = 16: MU = 65
            IF J = 8 THEN LU = 18: MU = 48
            IF J = 9 THEN LU = 15: MU = 13
            IF J > 0 THEN BEEP
        GOTO 5999

5100        IF J = 1 THEN 5105
            IF J = 2 THEN 5405
            IF J = 3 THEN 5205
            IF J = 4 THEN 5405
            IF J = 5 THEN 5405
            IF J = 6 THEN 5405
            IF J = 7 THEN 5350
            IF J = 8 THEN 5405
            IF J = 9 THEN 5105

'Assemble Temperature and Liquid Back Pressure Inputs.
5105    XX = XX + 1
        D1 = 1: D2 = 2: D3 = 3: D4 = 4: D5 = 5
        GOSUB 5500
        AA = II1 * 100 + II2 * 10 + II3 + II5 / 10
        GOTO 5999

'Assemble Mach Number Input.
5205    XX = XX + 1
        D1 = 1: D2 = 3: D3 = 4: D4 = 2: D5 = 5
        GOSUB 5500
        AA = II2 / 10 + II3 / 100 + II5 / 1000
        GOTO 5999

'Assemble Fan Speed Input.
5350    XX = XX + 1
        D1 = 1: D2 = 2: D3 = 3: D4 = 5: D5 = 4
        GOSUB 5500
        AA = II1 * 1000 + II2 * 100 + II3 * 10 + II5
        GOTO 5999
```

```
'Assemble Pressure, Reynolds Number, Injection Valve, Exhaust Valve, and Chord Inputs.
5405    XX = XX + 1
        D1 = 1: D2 = 2: D3 = 4: D4 = 3: D5 = 5
        GOSUB 5500
        AA = H1 * 10 + H2 + H3 / 10 + H5 / 100
        GOTO 5999

5500            IF J = 1 THEN COLOR 0, 3
                IF J = 2 THEN COLOR 0, 5
                IF J = 3 THEN COLOR 0, 6
                IF J = 4 THEN COLOR 0, 5
                IF J = 5 THEN COLOR 0, 3
                IF J = 6 THEN COLOR 0, 5
                IF J = 7 THEN COLOR 0, 6
                IF J = 8 THEN COLOR 0, 5
                IF J = 9 THEN COLOR 0, 2
                IF XX = D1 THEN H1 = VAL(A$): LOCATE LU, MU + D1: PRINT USING "#";H1
                IF XX = D2 THEN H2 = VAL(A$): LOCATE LU, MU + D2: PRINT USING "#";H2
                IF XX = D3 THEN H3 = VAL(A$): LOCATE LU, MU + D3: PRINT USING "#";H3
                IF XX = D4 THEN LOCATE LU, MU + D4: PRINT CHR$(46);
                IF XX = D5 THEN H5 = VAL(A$): LOCATE LU, MU + D5: PRINT USING "#";H5
        RETURN

'Set Temperature Input.
5600            IF J = 1 THEN 5620 ELSE 5660
5620    ST1 = AA
                IF ST1 > MAXT THEN ST1 = MAXT
                IF ST1 < MINT THEN ST1 = MINT
                IF ST1 < SAT1 THEN ST1 = SAT1
                STP = TT
        COLOR 14, 3
        LOCATE 3, 29: PRINT "  AUTO  ": PRINT ;
        AUTOT = 1
        LOCATE 5, 26: PRINT USING "###.#"; ST1

'Set Pressure Input.
5660            IF J = 2 THEN 5670 ELSE 5730
5670    SP = AA
                IF SP < MINP THEN SP = MINP
                IF SP > MAXP THEN SP = MAXP
        COLOR 14, 5
        LOCATE 3, 46: PRINT " AUTOP  ": PRINT ;
        LOCATE 6, 43: PRINT "      ": PRINT ;
        LOCATE 5, 43: PRINT USING "##.##"; SP
        AUTOP = 1
        AUTORE = 0

'Set Mach Number Input.
5730            IF J = 3 THEN 5740 ELSE 5750
5740    SM = AA
                IF SM < MINM THEN SM = MINM
                IF SM > MAXM THEN SM = MAXM
        COLOR 14, 6
        LOCATE 3, 64: PRINT " AUTO   ": PRINT ;
        LOCATE 6, 62: PRINT "      "
        AUTOM = 1
```

```
        LOCATE 5, 62: PRINT USING "#.###"; SM

'Set Reynolds Number Input.
5750            IF J = 4 THEN 5760 ELSE 5770
5760    SRE = AA
                IF SRE > MAXRE THEN SRE = MAXRE
                IF SRE < MINRE THEN SRE = MINRE
        COLOR 14, 5
        AUTORE = 1
        AUTOP = 1
        LOCATE 6, 43: PRINT USING "##.##"; SRE
        LOCATE 3, 46: PRINT " AUTORE ": PRINT ;

'Set Liquid Injection Valve Input.
5770            IF J = 5 THEN 5780 ELSE 5810
5780    LCMDS = AA
        COLOR 14, 3
        LOCATE 3, 29: PRINT " MANUAL ": PRINT ;
        LOCATE 5, 26: PRINT "      "
        LOCATE 6, 26: PRINT "      "
        AUTOT = 0

'Set Gas Exhaust Valve Input.
5810            IF J = 6 THEN 5820 ELSE 5860
5820    GCMDS = AA
        COLOR 14, 5
        LOCATE 3, 46: PRINT " MANUAL ": PRINT ;
        AUTOP = 0
        AUTORE = 0
        LOCATE 5, 43: PRINT "      ": PRINT ;
        LOCATE 6, 43: PRINT "      ": PRINT ;

'Set Fan Speed Input.
5860            IF J = 7 THEN 5870 ELSE 5935
5870    NCMDS = AA
                IF NCMDS > MAXN THEN NCMDS = MAXN
        COLOR 14, 6
        LOCATE 5, 62: PRINT "       "
        LOCATE 3, 64: PRINT " MANUAL ": PRINT ;
        LOCATE 6, 62: PRINT USING "####."; NCMDS
        AUTOM = 0

'Set Chord Length Input.
5935            IF J = 8 THEN 5937 ELSE 5943
5937    CHIN = AA
                IF CHIN < MINCHIN THEN CHIN = MINCHIN
                IF CHIN > MAXCHIN THEN CHIN = MAXCHIN
        COLOR 14, 5
        LOCATE 20, 48: PRINT USING "##.##"; CHIN
        CH = CHIN * .0254

'Set Liquid Back Pressure Input.
5943            IF J = 9 THEN 5944 ELSE 5950
5944    SLQSC = AA
                IF SLQSC > MAXLQSC THEN SLQSC = MAXLQSC
        COLOR 14, 2
        LOCATE 5, 13: PRINT USING "###.#"; SLQSC
```

```
5950            IF J = 0 THEN 5962
                IF J = 1 THEN COLOR 14, 3
                IF J = 2 THEN COLOR 14, 5
                IF J = 3 THEN COLOR 14, 6
                IF J = 4 THEN COLOR 14, 5
                IF J = 5 THEN COLOR 14, 3
                IF J = 6 THEN COLOR 14, 5
                IF J = 7 THEN COLOR 14, 6
                IF J = 8 THEN COLOR 14, 5
                IF J = 9 THEN COLOR 14, 2
        LOCATE LU, MU + 1: PRINT "   ": PRINT ;

5962            IF J > 0 THEN BEEP
                IF IE = 1 THEN COLOR 0, 6
                IF IE = 1 THEN BEEP: LOCATE 20, 60: PRINT "                ": PRINT ;
                IF PIE = 1 THEN LOCATE 21, 60: PRINT "            ": PIE = 0
                IF NIE = 1 THEN LOCATE 21, 60: PRINT "            ": NIE = 0
        J = 0
        XX = 0
        IE = 0

5999    RETURN

'Emergency Stop Subroutine.
6000    IE = 1
        COLOR 20, 8
        LOCATE 20, 60: PRINT " EMERGENCY STOP ": PRINT ;
        COLOR 4, 8
                IF PIE = 1 THEN LOCATE 21, 60: PRINT " Pressure Limit "
                IF NIE = 1 THEN LOCATE 21, 60: PRINT " Fan RPM Limit   "
        COLOR 4, 8
        AUTOT = 0
        AUTOP = 0
        AUTOM = 0
        AUTORE = 0
        LOCATE 3, 29: PRINT " MANUAL ": PRINT ;
        LOCATE 3, 46: PRINT " MANUAL ": PRINT ;
        LOCATE 3, 64: PRINT " MANUAL ": PRINT ;
        RETURN

'Simulation of 0.3-m TCT Dynamics.
7000    WG = 4375 * PP / TT
        WT = 3200!
        CM = (5.5 * TT - .008 * TT ^ 2) / 1000
        CP = 1.04
        CV = .75
        TMC = .28 * TT ^ 1.2 / PP ^ .7 / M ^ .7
        LMFL = ALQ * LF
        GMFL = (AGV1 + AGV2) * CGV * 2.725 * PP / SQR(TT)
                IF PP < 1.5 THEN GMFL = (AGV1 + AGV2) * PP / SQR(TT) * 21.8 * (2 - (1.5 / PP) ^ 1.7)
        WLHT = (TMWL - TT) * WT * CM / TMC
        HEAT = WLHT + FKW - LMFL * (121 + CV * TT) - GMFL * (CP - CV) * TT
        DPP = DEL * PP * HEAT / TT / WG / CV + DEL * (LMFL - GMFL) * PP / WG
        PPN = PP + DPP
                IF PPN > 6! THEN PPN = 6!
                IF PPN < 1! THEN PPN = 1!
```

```
        TMWLN = (DEL / (2 * TMC + DEL)) * (TT + TT1) + ((2 * TMC - DEL) / (2 * TMC + DEL)) *
        TMWL
        DFR1 = (1 / (DEL ^ 2 + 1.12 * DEL + .8))
        DFR2 = (DEL ^ 2) * (SRPM + SRPM1 * 2 + SRPM2) - FRPM * (2 * DEL ^ 2 - 1.6)
        DFR3 = DFR2 - FRPM1 * (DEL ^ 2 - 1.12 * DEL + .8)
        FRPMN = DFR1 * DFR3
        DTT = DEL * HEAT / WG / CV
        TTN = TT + DTT
        ALF1 = (ALN - .2) * 5
        PLO = 7.3 + ALF1 * 3 + ALQ * .7
        PLQN = DEL / (2 + DEL) * (PLO + PLO1) + (2 - DEL) / (2 + DEL) * PLQ
              IF PLQN < 7.3 THEN PLQN = 7.3
        PP1 = PP
        TT1 = TT
        TMWL1 = TMWL
        FRPM1 = FRPM
        SRPM2 = SRPM1
        PLO1 = PLO
        TT = TTN
              IF TT < SAT1 THEN TT = SAT1
        PP = PPN
        TMWL = TMWLN
        FRPM = FRPMN
        PLQ = PLQN
        RETURN

7500    CLS
        SYSTEM
```

Appendix C
Source Code Variables

Limits Input Keyboard Commands

A,a Accepts the tunnel maximum-minimum limits and starts the 0.3-m TCT controller.

D,d Delete the previous input keys shown on the screen and not yet executed.

M,m Input maximum Mach number limit. Maximum Mach number limit format is #.###, range to 0.15 to 0.995.

N,n Input maximum fan speed limit. Maximum fan speed limit format is ####., range is 0 to 5600 rpm.

P,p Input maximum tunnel total pressure limit. Maximum tunnel total pressure limit format is ##.##, range is 14.7 to 88.0 psia.

R,r Input maximum Reynolds number limit. Maximum Reynolds number limit format is ##.##, range is 1 to 50 million.

S,s Accepts the user set maximum-minimum limits and starts the 0.3-m TCT controller.

T,t Input minimum temperature limit. Minimum temperature limit format is ###.#, range is 77 to 340 K.

Controller Input Keyboard Commands

B,b Input LN_2 back pressure set point. LN_2 back pressure set point format is ###.#, range is LN_2 pressure to 150 psia.

C,c Input mean aerodynamic chord. Chord format is ##.##, range is 0.39 to 15.75 inches.

D,d Delete the previous input keys memorized on the screen and not yet executed.

G,g Input GN_2 discharge valve area. Takes the pressure controller and Reynolds number controller to manual control mode. GN_2 discharge valve area format is ##.##, range is 99.99%= full open to 0%=closed.

L,l Input LN_2 injection valve area. Takes the temperature controller to manual control mode. LN_2 injection valve area format is ##.##, range is 99.99%=full open to 0%=closed.

M,m Input Mach number set point. Takes fan speed to automatic Mach number control mode. Mach number set point format is #.###, range is 0.150 to 0.995.

N,n Input fan speed set point. Takes the fan speed to manual Mach number control mode. Fan speed set point format is ####., range is 0 to 5600 rpm.

P,p Input tunnel total pressure set point. Takes the controller to automatic pressure control mode and manual Reynolds number control mode. Total pressure set point format is ##.##, range is 14.7 to 88 psia.

R,r Input Reynolds number set point. Takes the pressure controller to automatic Reynolds number control mode by generating the required pressure set point. Reynolds number set point format is ##.##, range is 1 to 50 million.

T,t Input temperature set point. Takes the controller to automatic temperature control mode. Temperature set point format is ###.#, range is saturation temperature to 340 K.

Program Variables

AA Keyboard input function. Combines the integer inputs H1, H2, H3, and H5 with proper decimal scaling for use in program calculation.

AEMP Temporary data register output in digital to analog conversion routine to check status.

AGV1 Area of GN_2 discharge valve #1. Full open AGV1=1 and closed AGV1=0. AGV1=KPGS*(KPP*EP+KDP*(EP-2*EPM1+EPM2/2/DEL)+RIP

AGV1P Area of GN_2 discharge valve #1, previous cycle value.

AGV2 Area of GN_2 discharge valve #2. Full open AGV2=1 and closed AGV2=0. Valve #2 opens only when AGV1 > 90% open and starts closing when AGV1 < 70% open. Valve #2 moves at a rate of 1% per cycle.

ALF1 Simulator variable. Variable related to the area of LN_2 back pressure valve used in calculating the estimated LN_2 back pressure. ALF1=(ALN-.2)*5

ALN Area of LN_2 back pressure control valve. Full open ALN=1 and closed ALN=0. ALN=KPL*ELP+KDL*(ELP-ELPM1)*DEL+RIL

ALQ Area of LN_2 injection valve. Full open ALQ=1 and closed ALQ=0. ALQ=KTGS*(KPT*ET+KDT*(ET-2*ETM1+ETM2)/2/DEL)+RIT+FBF

ALQP Area of LN_2 injection valve, previous cycle value.

AUTOM — Mach number controller:
AUTOM=1 automatic Mach number control.
AUTOM=0 manual Mach number control.

AUTOP — Pressure controller:
AUTOP=1 automatic pressure control.
AUTOP=0 manual pressure control.

AUTORE — Reynolds number controller:
AUTORE=1 automatic Reynolds number control.
AUTORE=0 manual Reynolds number control.

AUTOT — Temperature controller:
AUTOT=1 automatic temperature control.
AUTOT=0 manual temperature control.

A$ — Input variable from the keyboard buffer.

BEMP — Temporary status register output in digital to analog conversion routine to check status.

BSD — Dummy calculation to create a time delay between resetting the digital to analog conversion and reading its status. Used only during initialization of digital to analog conversion.

B$ — Input variable from the keyboard buffer.

CEMP — Temporary status register output in digital to analog conversion subroutine.

C$ — Input variable from the keyboard buffer.

CGV — Coefficient for GN_2 exhaust valve.

CH — Aerodynamic chord in meters.

CHIN — Aerodynamic chord in inches. Loaded through the keyboard using the "C" command. Default value is CHIN=0.1800*0.0254.

CLQV — Flow coefficient of LN_2 injection valve.

CM — Simulator variable. Specific heat of tunnel metal.
CM=(5.5*TT-0.008*TT^2)/1000

CP — Simulator variable. Specific heat of nitrogen at constant pressure.

CRPM — Simulator variable. Fan speed in rpm with a minimum limit of 50 rpm.

CRM — Simulator variable. Variable used in calculating Mach number as a function of fan speed, pressure, and temperature.
CRM=2.77-CRPM/157/SQR(TT)*PP^0.035

CV — Simulator variable. Specific heat of nitrogen at constant volume.

DAC(1) — Control variable ALQ output.

DAC(2) — Control variable ALQ output.

DAC(3) — Control variable ALN output.

DAC(4) — Control variable AGV1 output.

DAC(5) — Control variable AGV2 output.

DAC(6) — Control variable SNRPM output.

DAC(7) — RPM/MACH controller output for rheostat control. DAC(7)=1 for normal operation. DAC(7)=0 for rheostat control.

D1 — Integer used to print properly the variable being loaded through the keyboard.

D2 — Integer used to print properly the variable being loaded through the keyboard.

D3 — Integer used to print properly the variable being loaded through the keyboard.

D4 — Integer used to print properly the variable being loaded through the keyboard.

D5 — Integer used to print properly the variable being loaded through the keyboard.

DEL — Cycle time step.

DFR1 — Simulator variable. Variable used in calculating the estimated fan speed rpm.
DFR1=(1/(DEL^2+1.12*DEL+0.8))

DFR2 — Simulator variable. Variable used in calculating the estimated fan speed rpm.

DFR2=(DEL^2)*(SRPM+SRPM1*2+SRPM2)-FRPM*(2*DEL^2-1.6)

DFR3 — Simulator variable. Variable used in calculating the estimated fan speed rpm.
DFR3=DFR2-FRPM1*(DEL^2-1.12*DEL+.8)

DLP — Difference in pressure across the screens of the tunnel settling chamber, (psia).

DLPC	Maximum safe screen pressure drop allowed. DLPC=0.47*PP*M*M/(MF)^6+0.02
DPP	Simulator variable. Estimated change in tunnel total pressure for one cycle. DPP=DEL*PP*HEAT/TT/WG/CV+DEL*(LMFL-GMFL)*PP/WG
DRPM	Fan speed in rpm with a minimum limit of 100 rpm.
DR1	Command output high integer buffer in digital to analog conversion routine, (12 bit).
DR2	Command output low integer buffer in digital to analog conversion routine, (4-bit).
DTT	Simulator variable. Estimated change in tunnel temperature for one cycle. DTT=DEL*HEAT/WG/CV
ELP	Error in LN_2 back pressure control, (atm). ELP=SLQ-PLQ
ELPM1	Error in LN_2 back pressure control, previous cycle value.
EM	Error in Mach number control. EM=SM-M
EN	Error in fan speed control, (rpm). EN=NCMDS1-FRPM
EP	Error in total pressure control, (atm). EP=PP-SPR
EPM1	Error in total pressure control, previous cycle value.
EPM2	Error in total pressure control, twice previous cycle value.
ET	Error in temperature control, (kelvin). ET=TT-ST
ETM1	Error in temperature control, previous cycle value.
ETM2	Error in temperature control, twice previous cycle value.
E(1)	Input millivolts from total pressure sensor.
E(2)	Input millivolts from static pressure sensor.
E(3)	Input millivolts from total temperature thermocouple.
E(4)	Input millivolts from the tunnel metal wall temperature thermocouple.
E(5)	Input millivolts from fan speed sensor.

E(6) Input millivolts from LN_2 pressure transducer.

E(7) Input millivolts from the screen pressure drop transducers.

E11 Output from analog to digital conversion of the tunnel total pressure for monitor display.

E12 Output from analog to digital conversion of the tunnel static pressure for monitor display.

FB Fan bias, an equivalent of FKW in LN_2 flow. FB=FKW/(121+TT)/LF

FBF Product of FB and FF. FBF=FB*FF*0.8

FF Feed forward logic integer. FF=1 the fan bias is fed forward as LN_2. FF=0 the fan bias is not fed forward as LN_2. The feed forward is off during tunnel warm-ups.

FKM Variable used in calculating the tunnel fan speed/test section coupling. FKM=1-0.5*M

FKW Estimated fan power released to gas, (kilowatts). FKW=100*PP*(FRPM/1000)^2.26/SQR(TTMP)

FL0 Logic flag. FL0=1 when the set limit maximum equals the tunnel minimum value. This condition is an unacceptable condition to run the tunnel controller. FL0=0 corresponds to acceptable tunnel set limits to run the tunnel controller.

FL1 Logic flag. FL1=1 when the error in the temperature control is less than 0.3 K. FL1=0 corresponds to a larger temperature control error.

FL2 Logic flag. FL2=1 when metal temperature is within 24 K of the tunnel temperature set point. FL2=0 corresponds to a larger temperature difference. FL1*Fl2=1 corresponds to the temperature of the tunnel at the temperature set point.

FL3 Logic flag. FL3=1 when the error in pressure control is less than 0.005 atm. FL3=0 corresponds to a larger pressure control error.

FL4 Logic flag. FL4=1 when the error in Mach number is less than 0.002. FL4=0 corresponds to a larger Mach number control error.

FL5 Logic flag. FL5=1 when the error in the LN_2 back pressure control is less than 0.4 atm. FL5=0 corresponds to a larger LN_2 back pressure error.

FL6	Logic flag. FL6=1 when the pressure difference across the tunnel screens is unsafe. FL6=0 corresponds to a safe pressure difference.
FL7	Logic flag. FL7=1 when the error in the Reynolds number control is less than 0.05. FL7=0 corresponds to a larger Reynolds number control error.
FL10	Logic flag. FL10=1 when a sensor fails. FL10=0 corresponds to normal operation. A sensor failure will cause an emergency stop.
FRPM	Fan speed in rpm.
FRPMN	Simulator variable. Estimated fan speed. FRPMN=DFR1*DFR3
GCMDS	GN_2 discharge valve area. Loaded through the keyboard using the "G" command.
GF1	Mass flow from the tunnel. GF1=2.725*CGV*PP/SQR(TT)
GMFL	Simulator variable. GN_2 exhaust mass flow.
GV1	Product of AGV1 and 100. GV1=AGV1*100
GV2	Product of AGV2 and 100. GV2=AGV2*100
H1	First integer of an input keyed during a variable input.
H2	Second integer of an input keyed during a variable input.
H3	Third integer of an input keyed during a variable input.
H5	Fourth integer of an input keyed during a variable input.
HEAT	Simulator variable. The total heat flow of the tunnel. HEAT=WLHT+FKW-LMFL*(121+CV*TT)-GMFL*(CP-CV)*TT
HI	High byte to digital to analog conversion buffer of the digital to analog conversion routine.
I	Integer cycle counter used to display the controller screen borders.
IE	Logic flag. IE=1 corresponds to an emergency stop. IE=0 corresponds to normal operation.
IK	Integer cycle counter used for output of digital to analog conversion variables.
IW	Integer cycle counter for integrating the wall temperature gradient.

J	Integer variable that identifies the input variables.
JD	Integer cycle counter for displaying certain variables on the screen.
JJ	Integer cycle counter for calculation of BSD.
K	Simulator variable. Integer cycle counter for a 600 cycle beep.
KDL	LN_2 back pressure control derivative gain.
KDP	Pressure control derivative gain.
KDT	Temperature control derivative gain.
KEN	A gradient term limiting the rpm of the fan speed.
KIL	LN_2 back pressure control integral gain.
KIM	Mach number control integral gain.
KIN	Fan speed control integral gain.
KIP	Pressure control integral gain.
KIT	Temperature control integral gain.
KMC	Constant corresponding to tunnel fan speed/test section coupling in the Mach number controller. KMC=520*SQR(TT)*FKM/PP^0.035
KMGS	Mach number control gain schedule function. KMGS=KMC*KMM
KMM	Mach number control gain.
KP	Pressure control gain.
KPGS	Pressure control gain schedule function. KPGS=750*KP/PP/SQR(TT)
KPL	LN_2 back pressure control proportional gain. KPL=0.2
KPM	Mach number control proportional gain.
KPN	Fan speed control proportional gain.
KPP	Pressure control proportional gain.
KPT	Temperature control proportional gain.

KRE	Constant used for evaluating Reynolds number related functions. KRE=63714*CH*M/TT^1.4/*(MF)^2.1
KT	Temperature control gain factor.
KTGS	Temperature control gain schedule function. KTGS=DRPM*SQR(PP)*KT/3.0/TT/LF
LC	Screen column number used in LOCATE statements.
LCMDS	LN_2 valve area. Loaded through the keyboard using the "L" command.
LDPQ	Difference between LN_2 back pressure and tunnel pressure. LDPQ=PLQ-PP
LF	LN_2 flow when LN_2 injection valve is full open. LF=CLQV*0.8676*SQR(LDPQ)
LMFL	Simulator variable. LN_2 injected mass flow. LMFL=ALQ*LF
LMT	Limiter for LN_2 flow. LMT=1-(TMWL-TT-MXT/2)*2/MXT
LO	An output driving the digital to analog conversion low buffer for final command. Low byte data register in digital to analog conversion routine.
LQ	Product of ALQ and 100. LQ=ALQ*100
LR	Screen row number used in LOCATE statements.
LU	Screen column number used in LOCATE statements.
M	Tunnel flow Mach number in the test section based on the difference between total and static pressure.
MAXCH	Maximum aerodynamic chord length limit in meters.
MAXCHIN	Maximum aerodynamic chord length limit in inches.
MAXLQSC	Tunnel maximum LN_2 back pressure.
MAXM	User set maximum Mach number limit. Must be less than MAXM1.
MAXM1	Tunnel maximum Mach number limit.
MAXN	User set maximum fan speed limit. Must be less than MAXN1.

MAXN1	Tunnel maximum fan speed limit.
MAXP	User set maximum pressure limit. Must be less than MAXP1.
MAXP1	Tunnel maximum pressure limit.
MAXRE	User set maximum Reynolds number limit. Must be less than MAXRE1.
MAXRE1	Tunnel maximum Reynolds number limit.
MAXSPR1	Maximum pressure set point estimated from the Reynolds number set point. Valid in automatic Reynolds number control. MAXSPR1=MAXP1/14.696
MAXT	Maximum temperature limit.
MAXT1	Tunnel maximum temperature limit.
MF	Isentropic function of the tunnel flow Mach number. MF=(1+0.2*M*M)
MINCH	Minimum aerodynamic chord length in meters.
MINCHIN	Minimum aerodynamic chord length in inches.
MINM	Minimum Mach number limit.
MINM1	Tunnel minimum Mach number limit.
MINN	Minimum fan speed limit.
MINN1	Tunnel minimum fan speed limit.
MINP	Minimum pressure limit.
MINP1	Tunnel minimum pressure limit.
MINRE	Minimum Reynolds number limit.
MINRE1	Tunnel minimum Reynolds number limit.
MINSPR1	Minimum pressure set point estimated from the Reynolds number set point. Valid in automatic Reynolds number control mode. MINSPR1=MINP1/14.696
MINT	User set minimum temperature limit. Must be greater than MINT1.
MINT1	Tunnel minimum temperature limit.

MU	Screen column number used in the LOCATE statements.
MXT	Maximum safe temperature difference allowed between tunnel gas and tunnel metal wall.
NCMDS	Fan speed set point. Loaded through the keyboard using the "N" command.
NCMDS1	Fan speed set point derived from NCMDS or other safety fan speed commands.
NIE	Logic flag. NIE=1 when the tunnel exceeds the fan speed rpm limit. NIE=1 will causes an emergency stop of the tunnel. NIE=0 corresponds to normal operation.
PIE	Logic flag. PIE=1 when the tunnel exceeds the total pressure limit. PIE=1 will cause an emergency stop of the tunnel. PIE=0 corresponds to normal operation.
PLO	Simulator variable. Variable used in calculating the estimated LN_2 back pressure. PLO=7.3+ALF1*3+ALQ*.7
PLO1	Simulator variable. Variable used in calculating the estimated LN_2 back pressure, previous cycle value.
PLQ	LN_2 back pressure, (atm).
PLQN	Simulator variable. Estimated LN_2 back pressure. PLQN=DEL/(2+DEL)*(PLO+PLO1)+(2-DEL)/(2+DEL)*PLQ
PLQUSCS	LN_2 back pressure, (psia). PLQUSCS=PLQ*14.696
PP	Tunnel total pressure, (atm).
PPN	Simulator variable. Estimated tunnel total pressure. PPN=PP+DPP
PP1	Simulator variable. Tunnel total pressure, previous cycle value.
PPUSCS	Tunnel total pressure, (psia). PPUSCS=PP*14.696
PS	Static pressure, (atm).
PSUSCS	Static pressure, (psia). PSUSCS=PS*14.696
RE	Flow Reynolds number based on aerodynamic chord. RE=KRE*PP
READ.WAIT	Digital to analog software read command.

RIL	LN_2 back pressure control integral error, (atm-sec). RIL=RILM1+ELP*KIL*DEL
RILM1	LN_2 back pressure control integral error, previous cycle value.
RIM	Mach number control integral error, (Mach-sec). RIM=RIMM1+KIM*EM*DEL*KMGS
RIMM1	Mach number control integral error, previous cycle value.
RIN	Fan speed control integral error, (rpm-sec). RIN=RINM1+EN*DEL*KIN
RINM1	Fan speed control integral error, previous cycle value.
RIP	Tunnel total pressure control integral error, (atm-sec). RIP=RIPM1+EP*KPGS*KIP*DEL
RIPM1	Tunnel total pressure control integral error, previous cycle value.
RIT	Temperature control integral error, (K-sec). RIT=RITM1+KIT*KTGS*DEL*ET
RITM1	Temperature control integral error, previous cycle value.
SAT	Nitrogen saturation temperature based on Jacobsens data. SAT=50+27.34*PS^0.296
SAT1	Nitrogen saturation temperature applied to tunnel static pressure. SAT1=SAT*MF
SLQ	LN_2 back pressure control set point, (atm). SLQ=SLQSC/14.696
SLQSC	LN_2 back pressure set point. Loaded through the keyboard using the "B" command.
SM	Mach number set point. Loaded through the keyboard using the "M" command.
SNRPM	Fan speed rheostat drive command normalized to one. SNRPM=SRPM/7500
SP	Tunnel total pressure set point. Loaded through the keyboard using the "P" command.
SPR	Tunnel total pressure set point, (atm). SPR=SP/14.696
SPR1	Tunnel total pressure set point estimated from the Reynolds number set point. Valid in automatic Reynolds number control mode. SPR1=SRE/KRE

SPRU — Tunnel total pressure set point estimated from the Reynolds number set point, (psia). SPRU=SPR*14.696

SRE — Reynolds number set point. Loaded through the keyboard using the "R" command.

SRPM — Fan speed command from control law. SRPM=EN*KPN+RIN

SRPM1 — Fan speed command from control law, previous cycle value.

ST — Tunnel use temperature set point. Based on a safe temperature for given tunnel conditions. Also for conditions when both TT and PP are high and the gas mass in the tunnel is inadequate to reach the final temperature set point.

ST1 — Tunnel temperature set point. Loaded through the keyboard using the "T" command, (kelvin).

STP — Tunnel use temperature set point, previous cycle value.

TMC — Simulator variable. Metal time constant of heat release. TMC=948/TT^.12/PP^.8/M^.8

TMWL — Tunnel metal wall temperature, (kelvin).

TMWLN — Simulator variable. Estimated tunnel metal wall temperature. TMWLN=(DEL/(2*TMC+DEL))*(TT+TT1)+((2*TMC-DEL)/(2*TMC+DEL))*TMWL

TMWL1 — Tunnel metal wall temperature, previous cycle value.

TT — Temperature of tunnel gas, (kelvin).

TTN — Simulator variable. Estimated tunnel gas temperature. TTN=TT+DTT

TT1 — Simulator variable. Temperature of tunnel gas, previous cycle value.

TTMP — Temperature of tunnel gas with a minimum limit of 80 K.

WG — Simulator variable. Mass of nitrogen gas in the tunnel. WG=4375*PP/TT

WLG — Rate of change for the tunnel metal wall temperature. Wall temperature gradient. WLG=WLG+TMWL-TMWL1

WLHT — Simulator variable. Rate of heat release from the metal walls in the tunnel. WLHT=(TMWL-TT)*WT*CM/TMC

WRITE.WAIT — Digital to analog software write command.

WT Simulator variable. Mass of the tunnel metal exposed to the tunnel flow.

XDLP Screen pressure drop transducer sensitivity.

XFRPM Fan speed sensor sensitivity.

XPLQ LN_2 back pressure sensor sensitivity.

XPP Tunnel total pressure transducer sensitivity.

XPS Static pressure transducer sensitivity.

XX Integer counter between 0 to 5. Represents the number of keyed in characters while inputting set points into the controller.

ZADIN Analog to Digital In software command.

ZASE.ADDRESS Sets the DT2801 series board base address.

ZATA.REGISTER Sets the data in register and data out register.

ZCLEAR Digital to analog software clear command.

ZDCHNL Analog to digital conversion channel number.

ZDGAIN Analog to digital conversion gain switch at software level.

ZEMP Data register output for digital to analog conversion.

ZIGH High byte of analog to digital conversion output.

ZOL# Analog to digital conversion output normalized to 5 volts.

ZOLT# Analog to digital conversion output as a 16 bit binary number.

ZOMMAND.REGISTER Register for the analog to digital commands.

ZOW Low byte of analog to digital conversion output.

ZSTOP Defines the hex value for a stop command, ZSTOP.

ZTATUS.REGISTER Register for the analog to digital status.

&H1 Hex value for ZCLEAR.

&H2 Hex value for WRITE.WAIT.

&H5	Hex value for READ.WAIT
&HC	Hex value for ZADIN.
&HF	Hex value for ZSTOP.
&H2EC	Base address on the PC/AT microcomputer for analog to digital conversion.
&H224	Port number on the PC/AT microcomputer for digital to analog conversion.
&H225	Port number on the PC/AT microcomputer for digital to analog conversion.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	December 1991	Contractor Report

4. TITLE AND SUBTITLE
The NASA Langley Research Center 0.3-Meter Transonic Cryogenic Tunnel Microcomputer Controller Source Code

5. FUNDING NUMBERS
C NAS1-18585
WU 505-59-54-07

6. AUTHOR(S)
W. Allen Kilgore and S. Balakrishna

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
ViGYAN Inc.
30 Research Drive
Hampton, VA 23666-1325

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Langley Research Center
Hampton, VA 23665-5225

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
NASA CR-189556

11. SUPPLEMENTARY NOTES
Langley Technical Monitor: Charles L. Ladson

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified - Unlimited

Subject Category 09

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

The 0.3-m Transonic Cryogenic Tunnel (TCT) microcomputer-based controller has been operating for several thousand hours in a safe and efficient manner. This document provides a complete listing of the source codes for the tunnel controller and tunnel simulator. Included also is a listing of all the variables used in these programs. Several changes made to the controller are described. These changes are to improve the controller ease of use and safety.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Cryogenic wind tunnel; Microcomputer controller	65
	16. PRICE CODE: A04

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	